Refine Search

Search Results -

Terms	Documents
L19 and authori$	1

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L20

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Thursday, September 23, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=EPAB,JPAB,DWPI,TDBD; THES=ASSIGNEE; PLUR=YES; OP=OR			
L20	L19 and authori$	1	L20
L19	L18 and (copy or copying or duplicat$ or copied)	20	L19
L18	(compar$ with hash$) and @pd<=20001113	191	L18
DB=USPT; THES=ASSIGNEE; PLUR=YES; OP=OR			
L17	L16 not l14	7	L17
L16	L15 and l12	9	L16
L15	380/280,281,30,255,251.ccls.	1124	L15
L14	L12 and l10	3	L14
L13	L12 and l3	0	L13
L12	L11 and hash$.clm.	84	L12
L11	L1 and (copy or copying or duplicat$ or copied).clm.	139	L11
L10	L9 and l4	3	L10
L9	L7 and (first adj hash$) and (second adj hash$)	11	L9
L8	L7 and l2	0	L8

h e b b cg b e e ch

L7	L1 and (copy or copying or duplicat$ or copied).clm.	139	L7
L6	L1 and (copy or copying or duplicat$ or copied)	864	L6
L5	L4 and l2	0	L5
L4	(compar$ same hash$) and authori$ and @ad<=20001113	711	L4
L3	L2 and l1	0	L3
L2	701/51,52,55,57,59,75,76,80,26.ccls.	1303	L2
L1	(compar$ with hash$) and @ad<=20001113	1363	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

L11: Entry 4 of 10 File: USPT May 15, 2001

DOCUMENT-IDENTIFIER: US 6233684 B1
** See image for Certificate of Correction **
TITLE: System for controlling the distribution and use of rendered digital works through watermaking

Application Filing Date (1):
19971010

Brief Summary Text (17):
A trusted rendering system for use in a system for controlling the distribution and use of digital works is disclosed. The currently preferred embodiment of the present invention is implemented as a trusted printer. However, the description of the invention herein applies to any rendering device. A trusted printer facilitates the protection of printed documents which have been printed from a system which controls the distribution and use of digital works. The system for controlling distribution and use of digital works provides for attaching persistent usage rights to a digital work. Digital works are transferred in encrypted form between repositories. The repositories are used to request and grant access to digital works. Such repositories are also coupled to credit servers which provide for payment of any fees incurred as a result of accessing or using a digital work.

Detailed Description Text (3):
Trusted rendering combines four elements: a usage rights language, encrypted on-line distribution, automatic billing for copies, and digital watermarks for marking copies that are rendered.

Detailed Description Text (5):
Encrypted Distribution. Digital works are distributed from trusted systems to trusted rendering devices via computer networks. To reduce the risk of unauthorized interception of a digital work during transmission, it is encrypted. Communication with the rendering system is by way of a challenge-response protocol that verifies the authorization and security of the rendering device.

Detailed Description Text (14):
Repositories communicate utilizing a set of repository transactions. The repository transactions embody a set of protocols for establishing secure session connections between repositories, and for processing access requests to the digital works. Note that digital works and various communications are encrypted whenever they are transferred between repositories.

Detailed Description Text (20):
FIG. 2 illustrates the repository 101 coupled to a credit server 201. The credit server 201 is a device which accumulates billing information for the repository 101. The credit server 201 communicates with repository 101 via billing transaction 202 to record billing transactions. Billing transactions are reported to a billing clearinghouse 203 by the credit server 201 on a periodic basis. The credit server 201 communicates to the billing clearinghouse 203 via clearinghouse transaction 204. The clearinghouse transactions 204 enable a secure and encrypted transmission of information to the billing clearinghouse 203.

Detailed Description Text (23):
FIG. 3 illustrates a printer as an example of a rendering system. Referring to FIG. 3, a printer system 301 has contained therein a printer repository 302 and a print device 303. It should be noted that the dashed line defining printer system 301 defines a secure system boundary. Communications within the boundary is assumed to be secure and in the clear (i.e. not encrypted). Depending on the security level, the boundary also represents a barrier intended to provide physical integrity. The printer repository 302 is an instantiation of the rendering repository 105 of FIG. 1. The printer repository 302 will in some instances contain an ephemeral copy of a digital work which remains until it is printed out by the print engine 303. In other instances, the printer repository 302 may contain digital works such as fonts, which will remain and be billed based on use. This design assures that all communication lines between printers and printing devices are encrypted, unless they are within a physically secure boundary. This design feature eliminates a potential "fault" point through which the digital work could be improperly obtained. The printer device 303 represents the printer components used to create the printed output.

Detailed Description Text (26):
The core repository services 411 comprise a set of functions required by each and every repository. For a trusted printer repository the core repository services will include engaging in a challenge response protocol to receive digital works and decryption of received digital data.

Detailed Description Text (29):
FIG. 5 is a flowchart illustrating the basic steps for creating a digital work that may be printed on a trusted printer so that the resulting printed document is also secure. Note that a number of well known implementation steps, e.g. encryption of digital works, have been omitted in order to not detract from the basic steps. First, a digital work is written, assigned usage rights including a print right which specifies watermark information and is deposited in repository 1, step 501. As will be described in more detail below, the assignment of usage rights is accomplished through the use of a rights editor. Deposit of the digital work into repository 1 is an indication that it is being placed into a controlled system. Next, repository 1 receives a request from repository 2 for access to the digital work, step 502 and repository 1 transfers a copy of the digital work to repository 2, step 503. For the sake of this example, it is assumed that a "trusted" session between repository 1 and repository 2 has been established. The challenge response protocol used in this interaction is described in the aforementioned U.S. Pat. No. 5,629,980 and thus no further discussion on the challenge response protocol is deemed necessary.

Detailed Description Text (30):
Repository 2 then receives a user request to print the digital work, step 504. Repository 2 then establishes a trusted session with a printer repository of is the printing system on which the digital work will be printed, step 505. The printer repository receives the encrypted digital work and determines if it has a print right, step 506. If the digital work has the print right, the printer repository decrypts the digital work and generates the watermark that will be printed on the digital work, step 507. The printer repository then transmits the decrypted digital work with the watermark to a printer device for printing, step 508. For example, the decrypted digital work may be a Postscripts file of the digital work.

Detailed Description Text (75):
The next steps for the digital work are that it is published and distributed. During this process, the digital work is protected by the encryption and other security systems that are employed and the rights travel with the document. Part of this process assures that any printer or workstation that has a copy of the document also has digital certificates which contain information identifying the trusted system, trusted printer, user, and so on (a process described in more

detail in issued U.S. Pat. No. 5,629,980).

Detailed Description Text (78):
The digital work is then decrypted and downloaded into the printer, step 1208. When the digital work is downloaded into the printer, part of the protocol is also to download the new "revised" glyph font, which now has characters corresponding to glyph boxes. This font looks more or less like the one that the publisher used in creating the document, except that the gray codes inside the font boxes now embed the data that the publisher wants to appear in the watermarks on the document.

Detailed Description Text (87):
To serve this market a "trust box" is provided which would be positioned in between the personal computer and the personal printer. The "trust box" would act as a print repository for the trusted printer system. This is a market where the purchase of such hardware would be justified by the convenience of digital delivery to the office, for those documents that publishers are unwilling to send in the clear (i.e. not encrypted). The cost of the trust box offsets either waiting for mail delivery or driving to another location to pick up trusted printer output.

Detailed Description Text (88):
FIG. 14 is an illustration of a trust box in a computer based system. Referring to FIG. 14, a personal computer 1401 is coupled to a network 1402. The personal computer 1401 itself is part of a trusted system in that it embodies a repository. The personal computer would receive digital works through the network 1402 (e.g. over the Internet). The personal computer 1401 is further coupled to trust box 1403. The communications between the repository contained in the personal computer 1301 and the trust box 1403 are encrypted for security purposes. Finally, the trust box 1403 is coupled to a printer 1404. The printer 1404 receives decrypted print streams for printing.

Detailed Description Text (93):
The distributor repository encrypts the document using DES or some other encryption code, step 1505. The encryption uses a key length that is compatible with requirements of security and legal constraints. The distributor repository encrypts the document key in an envelope signed by the public key of the printer box, step 1506. The distributor repository then sends the encrypted document and the envelope along to the consumers workstation.

Detailed Description Text (94):
The personal computer stores the encrypted document in its repository along with the envelope containing the key, step 1507.

Detailed Description Text (95):
At some point, the user decides to print the document. Using a print program, he issues a print request, step 1508. His personal computer contacts the trust box, retrieving its identity certificate encrypted in its public key, step 1509. It looks up the watermark information in certificates from the user, the computer itself, and the printer, step 1510. It downloads the watermark font to the printer through the trust box, step 1511.

Detailed Description Text (97):
The trust box contacts the printer. It decrypts the document giving the document key to a decryption means (e.g. an internal decryption chip), step 1513. It transmits the document to the printer in the clear, step 1514. Note that this is one place where a digital copy could be leaked, if a printer emulator was plugged into the print box to act like a printer. Presumably the security level of the trust box is set to a value that reflects the level of risk. The document is then printed, step 1515. Finally, the trust box reports billing to a Financial Clearinghouse, step 1516.

Detailed Description Text (100):
Security. The approach inhibits unauthorized photocopying through the use of watermarks. The approach inhibits digital copying by storing digital works in an encrypted form, where the consumer workstation does not have access to the key for decrypting the work.

Detailed Description Text (101):
Printer Limitations. The approach assumes that the user will plug the trusted print box into a standard printer. The printer is assumed to not have the capability of storing extra copies of the digital work.

Detailed Description Text (104):
Billing Variations. In the version presented here, the trusted print box has secure storage and programs for managing billing records. A simpler version of the approach would be to keep track of all billing on-line. For example, one way to do this would be to have the document printing start at the time that the customer orders it. In this variation, the document is still sent in encrypted form from the publisher, through the consumer's workstation, decrypted, and sent to the trusted print box, to the printer. The difference is that the trusted print box no longer needs to keep billing records and that the consumer must start printing the document at the time that the document is ordered.

Detailed Description Text (105):
Software-only Variation. Another variation on the desktop printing solution involves only software. The consumer/client purchases the work and orders the right to print it once. The on-line distributor delivers the work, encrypted, one page at a time. The consumer workstation has a program that decrypts the page and sends it to the printer with watermarks, and then requests the next page. At no time is a full decrypted copy available on the consumer's computer. The weak link in this approach is that the consumers computer does gain access to copies of pages of the work in the clear. Although this would be beyond the average consumer, it would be possible to construct software either to mimic runtime decryption software or modify it to save a copy of the work, one page at a time.

Detailed Description Text (118):
The distributor encrypts the document using DES or some other code, using a key length that is compatible with requirements of security and legal constraints, step 1702. It encrypts the document key in an envelope signed by the public key of server, step 1703. It sends the encrypted document to the server, step 1704.

Detailed Description Text (119):
Note that in some versions of this architecture, different levels of encryption and "scrambling" (less secure) are used on the document at different stages in the server. It is generally important to protect the document in all places where it might be accessed by outside parties. The use of lower security encoding is sometimes used to avoid potentially-expensive decryption steps at critical stages that would slow the operation of the printer.

Detailed Description Text (120):
In any event, the server stores the encrypted document, step 1705. At some point, the spooler gets ready to print the document. Before starting, it runs a process to create a new version of the glyph font that encodes the watermark data, step 1706. It looks up the required watermark information in its own certificates as well as certificates from the repository and user.

Current US Original Classification (1):
713/176

Other Reference Publication (12):
O'Conner, M.A., "New Distribution option of electronic publishers; Opener data

encryption and metering system for CD-Rom use; Column", CD-Rom Professional, vol. 7; No. 2, .COPYRGT.Mar. 1994, p. 134.

CLAIMS:

13. In a system for controlling the distribution and use of digital works, a method for providing a watermark on a rendered digital work comprising the steps of:

a) a digital work creator assigning a rendering right to said digital work and storing in a distribution repository, said rendering right specifying watermark information indicating information identifying a rendering event and rendering criteria that an instance of a rendering system must satisfy before the digital work can be rendered;

b) a user obtaining an encrypted version of said digital work from said distribution repository and storing in a user repository;

c) said user requesting that said digital work be rendered;

d) said user repository determining if said digital work has the appropriate rendering right;

e) if said digital work has the appropriate rendering right, said user repository communicating with a rendering repository to establish a trusted session;

f) said user repository transferring said digital work to said rendering repository;

g) said rendering repository gathering watermark information specified in said rendering right and determining that it meets the required rendering criteria;

h) said rendering repository encoding data for said watermark information;

i) said rendering repository decrypting said digital work and embedding said watermark information, to be transmitted for subsequent extraction of watermark information; and

j) said rendering repository transmitting said digital work with embedded watermark information to a rendering device for rendering.

19. In a system for controlling the distribution and use of digital works, a method for providing a watermark on a rendered digital work comprising the steps of:

a) a digital work creator assigning a rendering right to said digital work and storing in a distribution repository, said rendering right specifying criteria for a rendering system that must be satisfied before the digital work can be rendered and watermark information indicating information identifying a rendering event;

b) a user requesting a rendered version of said digital work be rendered on a user rendering system having a rendering repository;

c) said distribution repository determining if said user rendering system meets the specified criteria in said rendering right;

d) if said rendering system satisfies said specified criteria, said distribution repository encrypting said digital work and sending to said rendering repository;

e) said rendering repository gathering watermark information specified in said rendering right;

f) said rendering repository encoding data for said watermark information;

g) said rendering repository decrypting said digital work and embedding said watermark information, to be transmitted for subsequent extraction of watermark information; and

h) said rendering repository transmitting said digital work with embedded watermark information to a rendering device for rendering.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

Generate Collection Print

L11: Entry 4 of 10 File: USPT May 15, 2001

US-PAT-NO: 6233684
DOCUMENT-IDENTIFIER: US 6233684 B1
** See image for Certificate of Correction **

TITLE: System for controlling the distribution and use of rendered digital works through watermaking

DATE-ISSUED: May 15, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Stefik; Mark J.	Portola Valley	CA		
Petrie; Glen W.	Los Gatos	CA		
Okamoto; Steve A.	Torrance	CA		
Briggs; Nicholas H.	Palo Alto	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Contenaguard Holdings, Inc.	Wilmington	DE			02

APPL-NO: 08/ 948893 [PALM]
DATE FILED: October 10, 1997

PARENT-CASE:
This Application claims benefit of Provisional Application Ser. No. 60/039,275 filed Feb. 28, 1997.

INT-CL: [07] H04 N 7/167

US-CL-ISSUED: 713/176; 380/9, 380/54
US-CL-CURRENT: 713/176; 380/54

FIELD-OF-SEARCH: 380/51, 380/52, 380/55, 380/4, 380/54, 382/232, 713/176

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	3263158	July 1966	Bargen et al.	323/44
☐	4529870	July 1985	Chaum	235/380

h e b b g e e e f c e h e ge

☐	4658093	April 1987	Hellman	380/25
☐	4924378	May 1990	Hershey et al.	364/200
☐	4932054	June 1990	Chou et al.	380/4
☐	4937863	June 1990	Robert et al.	380/4
☐	4953209	August 1990	Ryder, Sr. et al.	380/23
☐	4961142	October 1990	Elliott et al.	364/408
☐	4977594	December 1990	Shear	380/4
☐	5010571	April 1991	Katznelson	380/40
☐	5014234	May 1991	Edwards, jr.	364/900
☐	5023907	June 1991	Johnson et al.	380/4
☐	5047928	September 1991	Wiedemer	364/406
☐	5050213	September 1991	Shear	380/25
☐	5058164	October 1991	Elmer et al.	380/50
☐	5103476	April 1992	Waite	380/4
☐	5113519	May 1992	Johnson et al.	395/600
☐	5146499	September 1992	Geffrotin	380/23
☐	5159182	October 1992	Eisele	235/492
☐	5191193	March 1993	Le Roux	235/379
☐	5204897	April 1993	Wyman	380/4
☐	5235642	August 1993	Wobber et al.	380/25
☐	5247575	September 1993	Sprague et al.	380/9
☐	5260999	November 1993	Wyman	384/4
☐	5263157	November 1993	Janis	395/600
☐	5263158	November 1993	Janis	395/600
☐	5291596	March 1994	Mita	395/600
☐	5339091	August 1994	Yamazaki et al.	345/104
☐	5432849	July 1995	Johnson et al.	380/21
☐	5438508	August 1995	Wyman	364/401
☐	5504814	April 1996	Miyahara	380/4
☐	5530235	June 1996	Stefik et al.	235/492
☐	5629980	May 1997	Stefik et al.	380/4
☐	5708717	January 1998	Alasia	380/51
☐	5745569	April 1998	Moskowitz et al.	380/4
☐	5748783	May 1998	Rhoads	382/232
☐	5761686	June 1998	Bloomberg	707/529
☐	5768426	June 1998	Rhoads	382/232
☐	5825892	October 1998	Braudaway et al.	380/51
	5943422	August 1999	Van Wie et al.	380/9

h e b b g e e e f c e h e ge

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0 332 707 A1	September 1989	EP	
2 236 604	April 1991	GB	
WO 92/20022	November 1992	WO	
WO 93/01550	January 1993	WO	

OTHER PUBLICATIONS

Press Release From Electronic Publishing Resources, Inc. (EPR) entitled "National Semiconductor and EPR Partner for Information Metering/Data Security Cards", dated Mar. 4, 1994.
Weber, R., "Digital Rights Management Technology", Oct. 1995.
Flasche, U. et al., "Decentralized Processing of Documents", Comput. & Graphics, vol. 10, No. 2, Great Britain, 1986, pp. 119-131.
Mori, R. et al., "Superdistribution: The Concept and the Architecture", The Transactions of the IEICE, vol. E73, No. 7, Tokyo, JP, 1990, pp. 1133-1146.
Weber, R., "Metering Technologies for Digital Intellectual Property", A Report to the International Federation of Reproduction Rights Organizations, Oct. 1994, pp. 1-29.
Clark, P.C. et al., "Bits: A Smartcard Protected Operating System", Communications of the ACM, vol. 37, No. 11, Nov. 1994, pp. 66-70 and 94.
Ross, P.E., "Data guard", Forbes, Jun. 6, 1994, p. 101.
Saigh, W.K., "Knowledge is Sacred", Video Pocket/Page Reader Systems, Ltd., 1992.
Kahn, R.E., "Deposit, Registration and Recordation In An Electronic Copyright Management System", Corporation for National Research Initiatives, Reston, Virginia Aug. 1992, pp. 1-19.
Hilts, P. et al., "Books While U Wait", Publishers Weekly, Jan. 1, 1994, pp. 48-50.

Strattner, A., "Cash register on a chip'may revolutionize software pricing and distribution; Wave Systems Corp. ", Computer Shopper, Vol. 14, No. 4, .COPYRGT.Apr. 1994, p. 62.
O'Conner, M.A., "New Distribution option of electronic publishers; Opener data encryption and metering system for CD-Rom use; Column", CD-Rom Professional, vol. 7; No. 2, .COPYRGT.Mar. 1994, p. 134.
Willett, S., "Metered PCs: Is your system watching you?; Wave Systems beta test new technology", InfoWorld, .COPYRGT.May 2, 1994, p. 84.
Linn, R.J., "Copyright and Information Services in the Context of The National Research and Education Network.sup.1 ", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 9-20.
Perritt, Jr., H.H., "Permissions Headers and Contract Law", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 27-48.
Upthegrove, L. et al., "Intellectual Property Header Descriptors: A Dynamic Approach", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 63-66.
Sirbu, M.A., "Internet Billing Service Design and Protype Implementations", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 67-80.
Simmel, S.S. et al., "Metering and Licensing of Resources: Kala's General Purpose Approach", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 81-110.
Kahn, R.E., "Deposit, Registration and Recordation in an Electronic Copyright Management System", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 111-120.

Tygar, J.D. et al., "Dyad: A System for Using Physically Secure Coprocessors", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1 Jan. 1994, pp. 121-152.
Griswold, G.N., "A Method for PRotecting Copyright on Networks", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 169-178.
Nelson, T.H., "A Publishing and Royalty Model for Networked Documents", IMA Intellectual Property Project Proceedings, vol. 1, Issue 1, Jan. 1994, pp. 257-260.

ART-UNIT: 212

PRIMARY-EXAMINER: Peeso; Thomas R.

ASSISTANT-EXAMINER: Jack; Todd

ATTY-AGENT-FIRM: Peabody LLP; Nixon Kaufman; Marc S.

ABSTRACT:

A trusted rendering system for use in a system for controlling the distribution and use of digital works. A trusted rendering system facilitates the protection of rendered digital works which have been rendered on a system which controls the distribution and use of digital works through the use of dynamically generated watermark information that is embedded in the rendered output. The watermark data typically provides information relating to the owner of the digital work, the rights associated with the rendered copy of the digital work and when and where the digital work was rendered. This information will typically aid in deterring or preventing unauthorized copying of the rendered work to be made. The system for controlling distribution and use of digital works provides for attaching persistent usage rights to a digital work. Digital works are transferred between repositories which are used to request and grant access to digital works. Such repositories are also coupled to credit servers which provide for payment of any fees incurred as a result of accessing a digital work.

28 Claims, 17 Drawing figures

Previous Doc Next Doc Go to Doc#

L10: Entry 3 of 3 File: USPT Aug 25, 1992

US-PAT-NO: 5142578
DOCUMENT-IDENTIFIER: US 5142578 A

TITLE: Hybrid public key algorithm/data encryption algorithm key distribution method based on control vectors

DATE-ISSUED: August 25, 1992

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Matyas; Stephen M.	Manassas	VA		
Johnson; Donald B.	Manassas	VA		
Le; An V.	Manassas	VA		
Prymak; Rostislaw	Dumfries	VA		
Wilkins; John D.	Somerville	VA		
Martin; William C.	Concord	NC		
Rohland; William S.	Charlotte	NC		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Business Machines Corporation	Armonk	NY			02

APPL-NO: 07/ 748407 [PALM]
DATE FILED: August 22, 1991

INT-CL: [05] H04L 9/30

US-CL-ISSUED: 380/21; 380/30, 380/49
US-CL-CURRENT: 380/280; 380/281, 380/30, 713/175

FIELD-OF-SEARCH: 380/21, 380/25, 380/30, 380/49

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4200770	April 1980	Hellman et al.	380/30
☐	4218582	August 1980	Hellman et al.	380/30

☐	4405829	September 1983	Rivest et al.	380/30
☐	4677670	June 1987	Henderson, Jr.	380/25
☐	4817144	March 1989	Citta et al.	380/21
☐	4850017	July 1989	Matyas, Jr. et al.	380/21
☐	4908861	March 1990	Brachtl et al.	380/30
☐	4918728	April 1990	Matyas et al.	380/21
☐	4924514	May 1990	Matyas et al.	380/25
☐	4924515	May 1990	Matyas et al.	380/25
☐	4941176	July 1990	Matyas et al.	380/21
☐	4993069	February 1991	Matyas et al.	380/21
☐	5007089	April 1991	Matyas et al.	380/49
☐	5073934	December 1991	Matyas et al.	380/30
☐	5103478	April 1992	Matyas et al.	380/21

OTHER PUBLICATIONS

R. W. Jones, "Some Techniques for Handling Encipherment Keys," ICL Technical Journal, Nov. 1982, pp. 175-188.
D. W. Davies & W. L. Price, "Security for Computer Networks," John Wiley & Sons, N.Y., 1984, Sec. 6.5, Key Management With Tagged Keys, pp. 168-172.
W. Diffie, et al., "Privacy and Authentication: An Introduction to Cryptography," Proc. of IEEE, vol. 67, No. 3, Mar. 1979; pp. 397-427.

ART-UNIT: 222

PRIMARY-EXAMINER: Cangialosi; Salvatore

ATTY-AGENT-FIRM: Hoel; John E.

ABSTRACT:

The patent describes a method and apparatus for securely distributing an initial Data Encryption Algorithm (DEA) key-encrypting key by encrypting a key record (consisting of the key-encrypting key and control information associated with that key-encrypting key) using a public key algorithm and a public key belonging to the intended recipient of the key record. The patent further describes a method and apparatus for securely recovering the distributed key-encrypting key by the recipient by decrypting the received key record using the same public key algorithm and private key associated with the public key and re-encrypting the key-encrypting key under a key formed by arithmetically combining the recipient's master key with a control vector contained in the control information of the received key record. Thus the type and usage attributes assigned by the originator of the key-encrypting key in the form of a control vector are cryptographically coupled to the key-encrypting key such that the recipient may only use the received key-encrypting key in a manner defined by the key originator.

The patent further describes a method and apparatus to improve the integrity of the key distribution process by applying a digital signature to the key record and by including identifying information (i.e., an originator identifier) in the control information of the key record. The integrity of the distribution process is enhanced by verifying the digital signature and originator identifier at the

☐			
☐	5613004	March 1997	Cooperman et al.
☐	5621797	April 1997	Rosen
☐	5629980	May 1997	Stefik et al.
☐	5633932	May 1997	Davis et al.
☐	5634012	May 1997	Stefik et al.
☐	5636292	June 1997	Rhoads
☐	5638443	June 1997	Stefik
☐	5638504	June 1997	Scott et al.
☐	5640546	June 1997	Gopinath et al.
☐	5655077	August 1997	Jones et al.
☐	5687236	November 1997	Moskowitz et al.
☐	5689587	November 1997	Bender et al.
☐	5692047	November 1997	McManis
☐	5692180	November 1997	Lee
☐	5710834	January 1998	Rhoads
☐	5715403	February 1998	Stefik
☐	5732398	March 1998	Tagawa
☐	5740549	April 1998	Reilly et al.
☐	5745604	April 1998	Rhoads
☐	5748763	May 1998	Rhoads
☐	5748783	May 1998	Rhoads
☐	5748960	May 1998	Fischer
☐	5754849	May 1998	Dyer et al.
☐	5758152	May 1998	LeTourneau
☐	5765152	June 1998	Erickson
☐	5768426	June 1998	Rhoads
☐	5774872	June 1998	Golden et al.
☐	5819263	October 1998	Bromley et al.
☐	5842173	November 1998	Strum et al.

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
9 004 79	December 1984	BE	
3803982A1	January 1990	DE	
0 84 441	July 1983	EP	
0128672	December 1984	EP	
A0135422	March 1985	EP	

0180460	May 1986	EP
0 370 146	November 1988	EP
0399822A2	November 1990	EP
0421409A2	April 1991	EP
0 456 386 A2	November 1991	EP
0 469 864 A3	February 1992	EP
0 469 864 A2	February 1992	EP
0 565 314 A2	October 1993	EP
0 593 305 A2	April 1994	EP
0 651 554 A1	May 1995	EP
0 668 695 A2	August 1995	EP
0 696 798 A1	February 1996	EP
0 695 985 A1	February 1996	EP
0715244A1	June 1996	EP
0715245A1	June 1996	EP
0715246A1	June 1996	EP
0715247A1	June 1996	EP
0715243A1	June 1996	EP
0 725 376	August 1996	EP
0749081A1	December 1996	EP
0 778 513 A2	June 1997	EP
0 795 873 A2	September 1997	EP
A2136175	September 1984	GB
2264796A	September 1993	GB
2294348	April 1996	GB
2295947	June 1996	GB
57-726	May 1982	JP
62-241061	October 1987	JP
64-68835	March 1989	JP
01-068835	March 1989	JP
02-242352	September 1990	JP
02-247763	October 1990	JP
02-294855	December 1990	JP
04-369068	December 1992	JP
05-181734	July 1993	JP
05-257783	October 1993	JP
05-268415	October 1993	JP
06-175794	June 1994	JP
215010	August 1994	JP
6225059	August 1994	JP
07-056794	March 1995	JP
07-084852	March 1995	JP
07-141138	June 1995	JP
07-200317	August 1995	JP
07-200492	August 1995	JP
07-244639	September 1995	JP
08-137795	May 1996	JP

08-152990	June 1996	JP
08-185298	July 1996	JP
WO A8502310	May 1985	WO
WO 85/03584	August 1985	WO
WO 90/02382	March 1990	WO
WO 92/06438	April 1992	WO
WO 92/22870	December 1992	WO
WO 93/01550	January 1993	WO
WO 94/01821	January 1994	WO
WO 94/03859	February 1994	WO
WO 94/06103	March 1994	WO
WO 94/16395	July 1994	WO
WO 94/18620	August 1994	WO
WO 94/22266	September 1994	WO
WO 94/27406	November 1994	WO
WO 95/14289	May 1995	WO
WO 96/00963	January 1996	WO
WO 96/06503	February 1996	WO
WO 96/05698	February 1996	WO
WO 96/03835	February 1996	WO
WO 96/13013	May 1996	WO
WO 96/21192	July 1996	WO
WO 97/03423	January 1997	WO
WO 97/07656	March 1997	WO
WO 97/32251	September 1997	WO
WO 97/48203	December 1997	WO

OTHER PUBLICATIONS

Applications Requirements for Innovative Video Programming; How to Foster (or Cripple) Program Development Opportunities for Interactive Video Programs Delivered on Optical Media; A Challenge for the Introduction of DVD (Digital Video Disc) (Oct. 19-20, 1995, Sheraton Universal Hotel, Universal City CA).
Argent Information Q&A Sheet, http://www.digital-watermark.com/, The DICE Company, 1995, 7 pages.
Arneke, David, et al., AT&T Encryption System Protects Information Services, News Release, Jan. 9, 1995, 1 page.
AT&T Technology, New Products Systems and Services, vol. 9, No. 4, pp. 16-19.
Baggett, Claude, Cable's Emerging Role in the Information Superhighway, Cable Labs, 13 slides.
Barassi, Theodore Sedgwick, The Cybernotary: Public Key Registration and Certification and Authentication of International Legal Transactions, 4 pages.
Barnes, Hugh, Memo to Henry LaMuth, Subject: George Gilder Articles, May 31, 1994.
Bart, Dan, Comments in the Matter of Public Hearing and Request for Comments on the International Aspects of the National Information Infrastructure, Aug. 12, 1994.
Baum, Michael, Worldwide Electronic Commerce: Law, Policy and Controls Conference, program details, Nov. 11, 1993.
Bisbey, II, et al., Encapsulation: An Approach to Operating System Security, Oct. 1973, pp. 666-675.
Blom, et al., Encryption Methods in Data Networks, Ericsson Technics, No. 2, 1978, Stockholm, Sweden.
Bruner, Rick E., Poweragent, Netbot Help Advertisers Reach Internet Shoppers,

(Document from Internet), Aug. 1997,.
Cable Television and America's Telecommunications Infrastructure, National Cable Television Association, Apr. 1993.
Caruso, Technology, Digital Commerce 2 plans for watermarks, which can bind proof of authorship to electronic works, New York Times, Aug. 1995.
Introducing The Workflow CD-ROM Sampler, Creative Networks, CD ROM, MCIMail: Creative Networks, Inc., Palo Alto, California.
Choudhury, et al., Copyright Protection for Electronic Publishing over Computer Networks, AT&T Bell Laboratores, Murray Hill, New Jersey 07974, Jun., 1994.
Clark, Tim, Ad Service Gives Cash Back, www.news.com, Aug. 4, 1997, 2 pages.
Codercard, Spec Sheet--Basic Coder Subsystem.
Intelligent Agents, Communications of the ACM, Jul. 1994, vol. 37, No. 7.
Communications of the ACM, Jun. 1996, vol. 39, No. 6.
Perspectives on the National Information Infrastructure: Ensuring Interoperability, Computer Systems Policy Project (CSSP), Feb. 1994.
Cunningham, Donna, et al., News Release, AT&T, AT&T, VLSI Technology Join to Improve Info Highway Security, Jan. 31, 1995, 3 pages.
Data Sheet, About the Digital Notary Service, Surety Technologies, Inc., 1994-95, 6 pages.
Dempsey, et al., D-Lib Magazine. Jul./Aug. 1996 The Warwick Metadata Workshop: A Framework for the Deployent of Resource Description, Jul. 15, 1966.
Denning, et al., Data Security, 11 Computing Surveys No. 3, Sep. 1979.
Diffie, Whitfield, et al., New Directions in Cryptography, IEEE Transactions on Information Theory, vol. 22, No. 6,, Nov. 1976, pp. 644-651.
Diffie, Whitfield, Privacy and Authentication: An Introduction to Cryptography, Proceedings of the IEEE, vol. 67, No. 3, Mar. 1979, pp. 397-427.
Digest of Papers, VLSI: New Architectural Horizons, Preventing Software Piracy With Crypto-Microprocessors, Robert M. Best, Feb. 1980, pp. 466-469.
DiscStore, Electronic Publishing Resources, 1991.
cgi@ncsa.uiuc.edu, CGI Common Gateway Interface, Document from Internet, 1996, 1 page.
DSP56000/DSP56001 Digital Signal Processor User's Manual, Motorola, 1990, p. 2-2.
Dusse, Stephen R., et al. A Cryptographic Library for the Motorola 56000 in Damgard, I.M., Advances in Cryptology-Proceedings Eurocrypt 90, Springer-Verlag, 1991, pp. 230-244.
Dyson, Esther, Intellectual Value, Wired Magazine, Jul. 1995, pp. 136-141 and 182-184.
A Publication of the Electronic Frontier Foundation, Effector Online, vol. 6, No. 6,, 8 pages, Dec. 6, 1993.
EIA and TIA White Paper on National Information Infrastructure, Electronic Industries Association and the Telecommunications Industry Association, Washington, D.C.
Electronic Currency Requirements, XIWT, Cross Industry Working Group.
Protecting Electronically Published Properties Increasing Publishing Profits, Electronic Publishing Resources, 1991.
www.ffly.com, What is Firefly? Firefly revision: 41.4, Firefly Network, Inc., 1995, 1996.
First CII Honeywell Bull International Symposium on Computer Security and Confidentiality, Jan. 26-28, 1981, Conference Text, pp. 1-21.
Framework for National Information Infrastructure Services, Draft, U.S. Department of Commerce, Jul. 1994.
Framework for National Information Infrastructure Services, NIST, Jul. 1994, 12 slides.
Garcia, D. Linda, Testimony Before a Hearing on Science, space and technology, May 26, 1994.
Gleick, James, Dead as a Dollar, The New York Times Magazine, Jun. 16, 1996, Section 6, pp. 26-30, 35, 42, 50, 54.
Intellectual Property and the National Information Infrastructure, a Preliminary Draft of the Report of the Working Group on Intellectual Property Rights, Green Paper, Jul. 1994.

Greguras, Fred, Copyright Clearances and Moral Rights, Softic Symposium '95, Nov. 30, 1995, 3 pages.
Guillou, L, Smart Cards and Conditional Access, pp. 480-490 Advances in Cryptography, Proceedings of EuroCrypt 84, Ed., Springer-Verlag, 1985.
Harman, Harry H., Modern Factor Analysis, Third Edition Revised, University of Chicago Press Chicago and London, Third revision published 1976.
Herzberg, Amir, et al., Public Protection of Software, ACM Transactions on Computer Systems, vol. 5, No. 4, Nov. 1987, pp. 371-393.
Hofmann, Jud, Interfacing the NII to User Homes, Electronic Industries Association, Consumer Electronic Bus Committee, 14 slides.
Holt, Stannie, Start-Up Promises User Confidentiality in Web Marketing Service, Info World Electric, (Document from Internet), Aug. 13, 1997.
Multimedia Mixed Object Envelopes Supporting a Graduated Fee Scheme via Encryption, IBM Technical Disclosure Bulletin, vol. 37, No. 03, Mar. 1994, Armonk, NY.
Transformer Rules for Software Distribution Mechanism-Support Products, IBM Technical Disclosure Bulletin, vol. 37, No. 04B, Apr. 1994, Armonk, NY.
IISP Break Out Session Report for Group No. 3, Standards Development and Tracking System.
Information Infrastructure Standards Panel: NII The Information Superhighway, Nations Bank--HGDeal--ASC X9, 15 pages.
Invoice? What is an Invoice? Business Week, Jun. 10, 1996.
Jiang, et al, A concept-Based Approach to Retrieval from an Electronic Industrial Directory, International Journal of Electronic Commerce, vol. 1, No. 1, 1996, pp. 51-72.
Jones, Debra, First Internet, Infomediary to Empower and Protect Consumers, PowerAgent Introducts, Document from Internet, Aug. 13, 1997, 3 pages.
Kelly, Kevin, E-Money, Whole Earth Review, 1993, pp. 40-59.
Kent, Protecting Externally Supplied Sotfware In Small Computers, (MIT/LCS/TR-255 Sep. 1980).
Kohntopp, M., Sag's durch die Blume, Apr. 1996, marit@schulung.netuse.de.
Kristol, et al., Anonymous Internet Mercantile Protocol, AT&T Bell Laboratories, Murray Hill, New Jersey, Draft: Mar. 17, 1994.
Lagoze, Carl, The Warwick Framework, A Container Architecture for Diverse Sets of Metadata, D-Lib Magazine, Jul./Aug. 1996.
Lanza, Mike, Electronic Mail, George Gilder's Fifth Article--Digital Darkhorse--Newspapers, Feb. 21, 1994.
Levy, Steven, That's What I Want, E-Money, Wired, Dec. 1994, 10 pages.
Low, et al., Anonymous Credit Cards and its Collusion Analysis, AT&T Bell Laboratories, Murray Hill, New Jersey, Oct. 10, 1994.
Low, et al., Anonymous Credit Cards, AT&T Bell Laboratories, Proceedings of the 2nd ACM Conference on Computer and Communications Security, Fairfax, Virginia, No. 2-4, 1994.
Low, et al., Document Marking and Identification using both Line and Word Shifting, AT&T Bell Laboratories, Murray Hill, New Jersey, Jul. 29, 1994.
Maclachlan, Malcolm, Debuts Spam-Free Marketing, TechWire, PowerAgent, (Document from Internet), Aug. 13, 1997, 3 pages.
Maxemchuk, Electronic Document Distribution, AT&T Bell Laboratories, Murray Hill, New Jersey 07974.
Micro Card--Micro Card Technologies, Inc., Dallas, Texas.
Milbrandt, E., Stenanography Info and Archive, 1996.
Mori, Ryoichi, et al., Superdistribution: The Concept and the Architecture, The Transactions of the Eieice, V, E73 (Jul. 1990), No. 7, Tokyo, Japan.
Mossberg, Walter S., Personal Technology, Threats to Privacy On-Line Become More Worrisome, Wall Street Journal, Oct. 24, 1996.
Negroponte, Nicholas, Electronic Word of Mouth, Wired, Oct. 1996, p. 218.
Negroponte, Nicholas, Some Thoughts on Likely and Expected Communications Scenarios: A Rebuttal, Telecommunications, Jan. 1993, pp. 41-42.
Neumann, et al., A Provably Secure Operating System: The System, Its Applications, and Proofs, Computer Science Laboratory Report CSL-116, Second Edition, SRI International May 1980.

Premenos Announces Templar 2.0--Next Generation Software for Secure Internet EDI, News Release, webmaster@templar.net, Jan. 17, 1996, 1 page.
The Document Company Xerox, Xerox Announces Software Kit For Creating Working Documents With Dataglyphs, News Release, Nov. 6, 1995, 13 pages.
The White House, Office of the President, Background on the Administration's Telecommunications Policy Reform Initiative, News Release, Jan. 11, 1994.
Nil, Architecture Requirements, XIWT.
Open System Environment Architectural Framework for National Information Infrastructure Services and Standards in Support of National Class Distributed Systems, Distributed System Engineering Program Sponsor Group, Draft 1.0, Aug. 5, 1994.
Pelton, Dr. Joseph N., Why Nicholas Negroponte is Wrong About the Future of Telecommunication, Telecommunications, Jan. 1993, pp. 35-40.
Portland Software's ZipLock, Internet information, Copyright Portland Software 1996-1997, 12 pages.
Proper Use of Consumer Information on the Internet, PowerAgent Inc., White Paper, (Document from Internet), Jun. 1997, 9 pages.
What the Experts are Reporting on PowerAgent, PowerAgent Press Releases, Document from Internet, Aug. 13, 1997, 6 pages.
What the Experts are Reporting on PowerAgent, PowerAgent Press Releases, Document from Internet, Aug. 4, 1997, 5 pages.
What the Experts are Reporting on PowerAgent, PowerAgent Press Releases, Document from Internet, Aug. 13, 1997, 3 pages.
The Future of Electronic Commerce, Premenos Corp. White Paper, A Supplement to Midrange Systems, Internet, webmaster@premenos.com, 4 pages.
National Semiconductor and EPR Partner For Information Metering/Data Security Cards, Press Release, Mar. 4, 1994.
Rankine, G., Thomas--A Complete Single-Chip RSA Device, Advances in Cryptography, Proceedings of CRYPTO 86, pp. 480-487 (A.M. Odlyzko Ed., Springer-Verlag 1987).
Reilly, Arthur K., Input to the `International Telecommunications Hearings, Standards committee T1-Telecommunications, ,` Panel 1: Component Technologies of the NII/GII.
Resnick, et al., Recommender Systems, Communications of the ACM, vol. 40, No. 3, Mar. 1997, pp. 56-89.
ROI Personal Library Software, 1987 or 1988.
ROI-Solving Critical Electronic Publishing Problems, Personal Library Software, 1987 or 1988.
Rose, Lance, Cyberspace and the Legal Matrix: Laws or Confusion?, 1991.
Rosenthal, Steve, Interactive Network: Viewers Get Involved, New Media, Dec. 1992, pp. 30-31.
Rosenthal, Steve, Interactive TV: The Gold Rush Is On, New Media, Dec. 1992, pp. 27-29.
Rosenthal, Steve, Mega Channels, New Media, Sep. 1993, pp. 36-46.
Rothstein, Edward, Technology, Connections, Making the Internet come to you, through `push` technology, The New York Times, Jan. 20, 1997, p. D5.
Rutkowski, Ken, Introduces First Internet `Infomediary` to Empower and Protect Consumers, PowerAgent, Tech Talk News Story, Document from Internet, Aug. 4, 1997.
Sager, Ira, Bits & Bytes, Business Week, Sep. 23, 1996, p. 142E.
Schlossstein, Steven, America: The GTs Comeback Kid, International Economy, Jun./Jul. 1993.
Sehurmann, Jurgen, Pattern Classification, A Unified View of Statistical and Neural Approaches, John Wiley & Sons, Inc., 1996.
Scnaumueller-Bichl, et al., A Method of Software Protection Based on the Use of Smart Cards and Cryptographic Techniques.
Serving the Community: A Public-Interest Vision of the National Information Infrastructure, Computer Professionals for Social Responsibility, Executive Summary.
Shear, Solutions for CD-ROM Pricing and Data Security Problems, CD ROM Yearbook 1988-1989 (Microsoft Press 1988 or 1989), pp. 530-533.
Smith, et al., Signed Vector Timestamps: A Secure Protocol for Partial Order Time,

CMU-93-116, School of Computer Science Carnegie Mellon University, Pittsburgh, Pennsylvania, Oct. 1991; version of Feb. 1993.
Special Report, The Internet: Fulfilling the Promise The Internet: Bring Order From Chaos; Lynch, Clifford, Search the Internet; Resnick, Paul, Filtering Information on the Internet; Hearst, Marti A., Interfaces for Searching the Web; Stefik, Mark, Trusted Systems; Scientific American, Mar. 1997, pp. 49-56, 62-64, 68-72, 78-81.
Stefik, Internet Dreams: Archetypes, Myths, and Metaphors, Letting Loose the Light: Igniting Commerce in Electronic Publication, Massachusetts Institute of Technology, 1996, pp. 219-253.
Stefik, Mark, Introduction to Knowledge Systems, Chapter 7, Classification, by Morgan Kaufmann Publishers, Inc.1995, pp. 543-607.
Stefik, Mark, Letting Loose the Light, Igniting Commerce in Electronic Publication, (1994, 1995) Palo Alto, California.
Stephenson, Tom, The Info Infrastructure Initiative: Data SuprHighways and You, Advanced Imaging, May 1993, pp. 73-74.
Sterling, Bruce, Literary freeware: Not for Commercial Use, remarks at Computers, Freedom and Privacy Conference IV, Chicago, Mar. 26, 1994.
Struif, Bruno, The Use of Chipcards for Electronic Signatures and Encryption in: Proceedings for the 1989 Conference on VSLI and Computer Peripherals, IEEE Computer Society Press, 1989, pp. 4/155-4/158.
Suida, Karl, Mapping New Applications Onto New Technologies, Security Services in Telecommunications Networks, Mar. 8-10, 1988, Zurich.
Templar Overview,: Premenos, Internet info@templar.net, 4 pages.
Templar Software and Services : Secure, Reliable, Standards-Based EDI Over the Internet, Premenos, Internet info@templar.net, 1 page.
The 1:1 Future of the Electronic Marketplace: Return to a Hunting and Gathering Society, 2 pages.
The Benefits of ROI For Database Protection and Usage Based Billing, Personal Library Software, 1987 or 1988.
The New Alexandria No. 1, Alexandria Institute, Jul.-Aug. 1986, pp. 1-12.
Tygar, et al., Cryptography: It's Not Just For Electronic Mail Anymore, CMU-CS-93-107, School of Computer Science Carnegie Mellon University, Pittsburgh, Pennsylvania, Mar. 1, 1993.
Tygar, et al., Dyad: A System for Using Physically Secure Coprocessors, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213.
Tygar, et al., Dyad: A System for Using Physically Secure Coprocessors, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, May 1991.
Valovic, T., The Role of Computer Networking in the Emerging Virtual Marketplace, Telecommunications, pp. 40-44.
Voight, Joan, Beyond the Banner, Wired, Dec. 1996, pp. 196, 200, 204.
Vonder Haar, Steven, PowerAgent Launches Commercial Service, Inter@ctive Week, (Document from Internet), Aug. 4, 1997.
Weber, Dr. Robert, Digital Rights Management Technologies, A Report to the International Federation of Reproduction Rights Organizations, Oct. 1995, pp. 1-49.

Weber, Dr. Robert, Digital Rights Management Technologies, Oct. 1995, 21 pages.
Weber, Metering, Technologies for Digital Intellectual Property, A Report to the International Federation of Reproduction Rights Organizations, Oct. 1994, pp. 1-29.

Weder, Adele, Life on The Infohighway, 4 pages.
Weingart,' Physical Security for the :ABYSS System, IBM, Thomas J. Watson Research Center, Yorktown Heights, New York 10598, 1987.
Weitzner, Daniel J., A Statement on EFF's Open Platform Campaign, Nov., 1993, 3 pages.
Wepin Store, Stenography Hidden Writing, Common Law, 1995.
White, ABYSS: A Trusted Architecture for Software Protection, IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598, 1987.
Is Advertising Really dead?, Wired 1.02, Part 2, 1994.
World Wide Web FAQ, How can I put an access counter on my home page?, 1996, 1 page.

XIWT Cross Industry Working Team, 5 pages, Jul. 1994.
Yee, Using Secure Coprocessors, CMU-CS-94-149, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213.

ART-UNIT: 362

PRIMARY-EXAMINER: Buczinski; Stephen C.

ATTY-AGENT-FIRM: Finnegan, Henderson, Farabow, Garrett & Dunner L.L.P.

ABSTRACT:

Secure computation environments are protected from bogus or rogue load modules, executables and other data elements through use of digital signatures, seals and certificates issued by a verifying authority. A verifying authority--which may be a trusted independent third party--tests the load modules or other executables to verify that their corresponding specifications are accurate and complete, and then digitally signs the load module or other executable based on tamper resistance work factor classification. Secure computation environments with different tamper resistance work factors use different verification digital signature authentication techniques (e.g., different signature algorithms and/or signature verification keys)--allowing one tamper resistance work factor environment to protect itself against load modules from another, different tamper resistance work factor environment. Several dissimilar digital signature algorithms may be used to reduce vulnerability from algorithm.

14 Claims, 20 Drawing figures

Previous Doc Next Doc Go to Doc#

L3: Entry 1 of 1 File: USPT Oct 10, 2000

US-PAT-NO: 6131162
DOCUMENT-IDENTIFIER: US 6131162 A

TITLE: Digital data authentication method

DATE-ISSUED: October 10, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Yoshiura; Hiroshi	Kawasaki			JP
Takaragi; Kazuo	Ebina			JP
Sasaki; Ryoichi	Fujisawa			JP
Susaki; Seiichi	Yokohama			JP
Toyoshima; Hisashi	Hachioji			JP
Saito; Tsukasa	Tokyo			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Hitachi Ltd.	Tokyo			JP	03

APPL-NO: 09/ 090419 [PALM]
DATE FILED: June 4, 1998

PARENT-CASE:
CROSS REFERENCE TO RELATED APPLICATIONS This application is related to application Ser. No. 09/385,638, filed Aug. 27, 1999, entitled "Method of Generating Authentication Enabled Electronic Data", by Y. Nagai et al; and application Ser. No. 09/371,526, filed Aug. 10, 1999, entitled "Method of Appending Information to Image and Method of Extracting Information from Image", by H. Yoshiura et al.

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	9-148061	June 5, 1997
JP	9-348860	December 18, 1997

INT-CL: [07] H04 L 9/32, H04 L 9/28, H04 L 9/30

US-CL-ISSUED: 713/176; 713/170, 713/181, 705/57, 380/28, 380/30
US-CL-CURRENT: 713/176; 380/28, 380/30, 705/57, 713/170, 713/181

FIELD-OF-SEARCH: 380/30, 380/28, 380/202, 380/279, 380/283, 705/51-58, 713/150, 713/155, 713/162, 713/168, 713/170, 713/176, 713/180, 713/181

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL | Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5530759	June 1996	Braudaway et al.	380/54
☐	5872848	February 1999	Romney et al.	380/25
☐	5892904	April 1999	Atkinson et al.	395/187.01
☐	5898779	April 1999	Squilla et al.	380/23
☐	5960081	September 1999	Vynne et al.	380/10

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0855829	July 1989	EP	
0590884	April 1994	EP	
0705025	April 1996	EP	
0854633	July 1998	EP	
0859503	August 1998	EP	
53-148918	December 1978	JP	
6431198	February 1989	JP	

OTHER PUBLICATIONS

F. Rouaix, "A Web Navigator with Applets in Caml", 1996, Published by Elsevier Science B.V., Computer Networks and ISDN Systems 28, pp. 1365-1371.
S. Anderson, et al, "Sessioneer: Flexible Session Level Authentication with off the shelf servers and clients", 1995 Elsevier Science B.V., Computer Networks and ISDN Systems 27, pp. 1047-1053.
W. Bender, "Techniques for Data Hiding", IBM Systems Journal, vol. 35, No. 3 & 4, 1996, pp. 313-330.
N. Komatsu, et al, A Proposal on Digital Watermark in Document Image Communication and its Application to Realizing a Signature, Electronics and Communications in Japan 73(1990) May, No. 5, Part I, New York, US, pp. 22-33.
B. Schneier, Applied Cryptography 1996, John Wiley & Sons, US New York, pp. 39-41.
Sasaki, et al, Security Technology for Open Networks, Hitachi Review, JP, Hitachi, Ltd., Tokyo, vol. 46, No. 4, pp. 197-202.
M. Schneider et al, A Robust Content Based Digital Signature for Image Authentication, Proceedings of the International Conference on Image Processing, US, New York, IEEE, pp. 227-230.
W. Bender, Techniques for Data Hiding, IBM Systems Journal, vol. 35, No. 3 & 4, 1996, pp. 313-336.
Eiji Okamoto, Ango Riron Nyumon (Introduction to Cryptography), Kyoritsu Shuppan Co., Ltd., 1993, pp. 133-137.
Bruce Schneier, Applied Cryptography, 2.sup.nd Ed., John Wilsy & Sons, Inc., 1996, pp. 39-41.
Nikkei Electronics., No. 683, 1997, pp. 99-107.
Opendesign., Apr., 1996, pp. 4-22.
Opendesign., Apr. 1996, pp. 40-78.
Jyohoshori (Information Processing), Jyohoshori Gakkai (Information

Processing Society of Japan), vol. 38, No. 9, 1997, pp. 752-810.

ART-UNIT: 277

PRIMARY-EXAMINER: Swann; Tod R.

ASSISTANT-EXAMINER: Darrow; Justin T.

ATTY-AGENT-FIRM: Antonelli, Terry, Stout & Kraus, LLP

ABSTRACT:

This invention provides a method for identifying a purchaser who purchased content from which an illegal copy was produced. A provider system encrypts a content purchased by the purchaser using a public key of a purchaser system and sends the encrypted content to the purchaser system. The purchaser system creates a digital signature of the content with the use of a private key of its own and embeds the created digital signature into the received content. When an illegal copy is found, the provider system verifies the digital signature, embedded in the illegal copy as a digital watermark, to identify the purchaser who purchased the content from which the illegal copy was produced.

63 Claims, 29 Drawing figures

Previous Doc Next Doc Go to Doc#

L3: Entry 1 of 1 File: USPT Oct 10, 2000

DOCUMENT-IDENTIFIER: US 6131162 A
TITLE: Digital data authentication method

Brief Summary Text (22):
This technique embeds the identification of the contents purchaser into the contents in the form of a digital watermark. When illegally copied contents are seized, the embedded information is extracted to identify the person (that is, the purchaser) who produced the illegal copy.

Brief Summary Text (23):
The basic procedure for embedding purchaser's identification information is as follows: (1) The provider (contents provider) assigns a unique number to a contents purchaser. (2) The provider embeds the number of the contents purchaser into the contents in the form of a digital watermark. (3) When illegally-copied contents are found and seized, the provider or inspection division extracts the number from the contents to identify the purchaser. (4) The penalty is imposed on the purchaser for illegal copy or for lending the contents to a person who produced the illegal copy.

Brief Summary Text (37):
For example, with the illegal copy prevention technique described above, a number embedded in the illegally-copied contents cannot always be used as a proof that the illegally-copied contents were purchased by the purchaser corresponding to that number. That is, because the number was given by the provider one-sidedly, the purchaser may insist that the number found in the copy is not the one assigned to him or her.

Brief Summary Text (44):
On the other hand, the digital signature technique is cumbersome because digital data as well as the digital signatures associated with the digital data must be managed as a pair. In addition, digital signatures, which can be separated from digital data much easier than digital watermarks, cannot be used for preventing illegal copies.

Brief Summary Text (48):
Take the logo mark of a credit card company for example. Imagine that an illegal user copies the logo mark of a credit card company from the Web page of a legal agent of the company, pastes it into an appropriate location of the Web page of the agent owned by the illegal user, and then stores the Web page in the WWW server so that any computer may access it. In this case, a consumer may judge, from the logo mark of the credit card company contained in the Web page of the agent owned by the illegal user, that the agent is legal and may send data necessary for settlement, such as a credit card number, to that WWW server. As a result, the illegal user is able to obtain the credit number of the consumer illegally and make an illegal profit.

Brief Summary Text (56):
To achieve the above object, this invention is an embed-in-content information

processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of embedding a digital signature into the content such that the digital signature cannot be separated from the content without using a predetermined rule, the digital signature being created by encrypting an n-bit hash value using a private key in accordance with a public key cipher system used by a first content-handling person, the n-bit hash value being obtained by evaluating the content with a first hash function; and sequentially repeating digital signature embedding for a second person to a k-th content-handling person, wherein, for an i-th content-handling person (i is an integer between 2 and k), the content into which the digital signatures of the first to an (i-1) contenthandling persons are embedded is evaluated with a second hash function, wherein a resulting n/2-bit hash value is encrypted using the private key of the i-th content-handling person to generate the digital signature of the i-th content-handling person, and wherein the digital signature of the i-th content-handling person is embedded into the content in which the digital signatures from the first to the (i-1)th persons are already embedded such that the digital signature of the i-th content-handling person cannot be separated from the content without using a predetermined rule.

Detailed Description Text (5):
The first embodiment explains an example of authentication of the relation between digital data and an individual/organization. More specifically, the embodiment explains an example of authentication of the relation between a content, one type of digital data, and a content purchaser, one type of individual/organization, in order to prevent the content from being copied illegally. However, it should be noted that the individual/organization need not always be a content purchaser. Depending upon the situation in which this embodiment is used, the first embodiment may be modified such that the individual/organization is a content copyright holder, a content vendor, a content wholesaler, or some other related person. In addition, in this embodiment and in the second and third embodiment that will be described later, the content is assumed to be image data. These embodiments may also be modified so that the content may contain other types of data, such as text data, drawing data, audio data, or video data.

Detailed Description Text (25):
When the illegally-copied content in which the digital signature is embedded is seized, the provider system 100 performs the following to identify the purchaser who created the illegal copy.

Detailed Description Text (26):
That is, as shown in FIG. 6, the controlling module 112 of the provider system 100 works with the input/output module 111 to store the illegally-copied content in the storage module 120 and then tells the signature extracting module 113 to extract the digital signature from the illegally-copied content (step 601). Note that the storage module 120 of the provider system 100 contains the original content (with no digital signature embedded) of the illegally-copied content. This allows the signature extracting module 113 to find the difference between the original content and the illegally-copied content and therefore to extract the digital signature. If it is possible, the digital signature may be extracted according to the rule by which the digital signature was embedded into the content.

Detailed Description Text (27):
Next, the controlling module 112 tells the signature verifying module 114 to verify the digital signature (step 602). To do so, the signature verifying module 114 decrypts the extracted digital signature using the verification key 122 of a user stored in the storage module 120 and compares the resulting value with the hash value obtained by evaluating the original content in the storage module 120 with the use of the same one-way hash function as that used by the purchaser system 200. If the rule used by the purchaser system 200 to embed the digital signature into the content is known only to the provider and if the digital signature may be

removed from the content according to that rule, the content from which the digital signature is removed may be used instead of the original content.

Detailed Description Text (138):
Therefore, in the fourth embodiment, if an illegal vendor copies the mark from the Web page of a legal vendor into his own Web page, the validity of the mark cannot be checked during the validity check because the mark management DB 1123 managed by the mark manager does not contain a record indicating that the mark was sent to the Web page of the illegal user. As a result, the consumer 1100 who browses the vendor's Web page can check the validity of the information indicated by the mark pasted in the Web page.

Detailed Description Text (168):
Therefore, when an illegal user copies a signature-containing mark from the Web page of an agent and pastes it into his own Web page, the URL of the Web page of the illegal user does not match the URL contained in the signature and so the mark cannot be validated during validity check processing. As a result, the consumer 1100 browsing the Web page of the vendor 1110 can validate the information indicated by the mark pasted in the Web page.

Detailed Description Text (199):
That is, as shown in FIG. 24, the terminal 1101 first extracts a mark 2407 from a Web page 2406 to check its validity (step 2401) and extracts a hash value 2408 embedded in the extracted mark 2407 as a digital watermark (step 2402). The terminal 1101 also calculates a hash value 2409 of the Web page data except the part related to the mark whose validity is to be checked (step 2403) and compares the calculated hash value 2409 with the hash value 2408 extracted from the mark (step 2404). If they match, the terminal 1101 displays a message stating that the mark was validated on the display unit 1102; if they do not match, the terminal 1101 displays a message stating that the mark was not validated on the display unit 1102 (step 2405).

Detailed Description Text (215):
That is, as shown in FIG. 29, the terminal 1800a first gets a public key 2910 of the mark management organization 1121 from the public key DB 1801. Then, the terminal 1800a extracts a mark 2908 from a Web page 2907 to check its validity (step 2901), extracts a digital signature 2909 embedded in the extracted mark 2908 as a digital watermark (step 2902), and decrypts the extracted digital signature using the public key 2910 of the mark management organization 1121 to get a hash value 2911 (step 2903). The terminal 1800a also calculates a hash value 2912 of the Web page data except the part related to the mark 2908 whose validity is to be checked (step 2904), and compares the calculated hash value 2912 with the hash value 2911 generated by decrypting the digital signature extracted from the mark 2908 (step 2905). If they match, the terminal 1800a displays a message on the display unit 1102 stating that the mark was validated; if they do not match, the terminal 1800a displays a message staging that the mark was not validated (step 2906).

Detailed Description Text (222):
That is, in the sixth embodiment, the consumer terminal extracts the mark to be validated from the Web page, and sends the extracted mark and a validity check request to the mark management server. In the seventh and eighth embodiments, the consumer terminal sends Web page data containing the mark and the validity check request to the mark management server. On the display unit of the consumer terminal there is displayed a successful or an unsuccessful validity check message sent back from the mark management server. On the other hand, upon receiving a validity check request, the mark management server performs the validity check on the mark in the same way as the consumer terminal performs in the sixth to eighth embodiments. In the sixth embodiment, the mark management server extracts information embedded in the mark sent with the request. If this information matches the information

embedded by the mark management server, it sends a successful validity message to the consumer terminal; if not, it sends an unsuccessful validity check message to the consumer terminal. In the seventh embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the hash value embedded in the mark as the digital watermark, calculates the hash value of the Web page except the area related to the mark to be validated, and compares this value with the hash value extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal. In the eighth embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the digital signature embedded in the extracted mark as the digital watermark, and extracts the hash value by decrypting the digital signature with a public key of the mark management organization. The mark management server calculates the hash value of the Web page data except the area related to the mark to be validated, and compares this value with the hash value generated by decrypting the digital signature extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal.

Previous Doc Next Doc Go to Doc#

L2: Entry 1 of 1 File: USPT Oct 10, 2000

DOCUMENT-IDENTIFIER: US 6131162 A
TITLE: Digital data authentication method

Brief Summary Text (22):
This technique embeds the identification of the contents purchaser into the contents in the form of a digital watermark. When illegally copied contents are seized, the embedded information is extracted to identify the person (that is, the purchaser) who produced the illegal copy.

Brief Summary Text (23):
The basic procedure for embedding purchaser's identification information is as follows: (1) The provider (contents provider) assigns a unique number to a contents purchaser. (2) The provider embeds the number of the contents purchaser into the contents in the form of a digital watermark. (3) When illegally-copied contents are found and seized, the provider or inspection division extracts the number from the contents to identify the purchaser. (4) The penalty is imposed on the purchaser for illegal copy or for lending the contents to a person who produced the illegal copy.

Brief Summary Text (37):
For example, with the illegal copy prevention technique described above, a number embedded in the illegally-copied contents cannot always be used as a proof that the illegally-copied contents were purchased by the purchaser corresponding to that number. That is, because the number was given by the provider one-sidedly, the purchaser may insist that the number found in the copy is not the one assigned to him or her.

Brief Summary Text (44):
On the other hand, the digital signature technique is cumbersome because digital data as well as the digital signatures associated with the digital data must be managed as a pair. In addition, digital signatures, which can be separated from digital data much easier than digital watermarks, cannot be used for preventing illegal copies.

Brief Summary Text (48):
Take the logo mark of a credit card company for example. Imagine that an illegal user copies the logo mark of a credit card company from the Web page of a legal agent of the company, pastes it into an appropriate location of the Web page of the agent owned by the illegal user, and then stores the Web page in the WWW server so that any computer may access it. In this case, a consumer may judge, from the logo mark of the credit card company contained in the Web page of the agent owned by the illegal user, that the agent is legal and may send data necessary for settlement, such as a credit card number, to that WWW server. As a result, the illegal user is able to obtain the credit number of the consumer illegally and make an illegal profit.

Detailed Description Text (5):
The first embodiment explains an example of authentication of the relation between

digital data and an individual/organization. More specifically, the embodiment explains an example of authentication of the relation between a content, one type of digital data, and a content purchaser, one type of individual/organization, in order to prevent the content from being copied illegally. However, it should be noted that the individual/organization need not always be a content purchaser. Depending upon the situation in which this embodiment is used, the first embodiment may be modified such that the individual/organization is a content copyright holder, a content vendor, a content wholesaler, or some other related person. In addition, in this embodiment and in the second and third embodiment that will be described later, the content is assumed to be image data. These embodiments may also be modified so that the content may contain other types of data, such as text data, drawing data, audio data, or video data.

Detailed Description Text (25):
When the illegally-copied content in which the digital signature is embedded is seized, the provider system 100 performs the following to identify the purchaser who created the illegal copy.

Detailed Description Text (26):
That is, as shown in FIG. 6, the controlling module 112 of the provider system 100 works with the input/output module 111 to store the illegally-copied content in the storage module 120 and then tells the signature extracting module 113 to extract the digital signature from the illegally-copied content (step 601). Note that the storage module 120 of the provider system 100 contains the original content (with no digital signature embedded) of the illegally-copied content. This allows the signature extracting module 113 to find the difference between the original content and the illegally-copied content and therefore to extract the digital signature. If it is possible, the digital signature may be extracted according to the rule by which the digital signature was embedded into the content.

Detailed Description Text (27):
Next, the controlling module 112 tells the signature verifying module 114 to verify the digital signature (step 602). To do so, the signature verifying module 114 decrypts the extracted digital signature using the verification key 122 of a user stored in the storage module 120 and compares the resulting value with the hash value obtained by evaluating the original content in the storage module 120 with the use of the same one-way hash function as that used by the purchaser system 200. If the rule used by the purchaser system 200 to embed the digital signature into the content is known only to the provider and if the digital signature may be removed from the content according to that rule, the content from which the digital signature is removed may be used instead of the original content.

Detailed Description Text (138):
Therefore, in the fourth embodiment, if an illegal vendor copies the mark from the Web page of a legal vendor into his own Web page, the validity of the mark cannot be checked during the validity check because the mark management DB 1123 managed by the mark manager does not contain a record indicating that the mark was sent to the Web page of the illegal user. As a result, the consumer 1100 who browses the vendor's Web page can check the validity of the information indicated by the mark pasted in the Web page.

Detailed Description Text (168):
Therefore, when an illegal user copies a signature-containing mark from the Web page of an agent and pastes it into his own Web page, the URL of the Web page of the illegal user does not match the URL contained in the signature and so the mark cannot be validated during validity check processing. As a result, the consumer 1100 browsing the Web page of the vendor 1110 can validate the information indicated by the mark pasted in the Web page.

Detailed Description Text (199):

That is, as shown in FIG. 24, the terminal 1101 first extracts a mark 2407 from a Web page 2406 to check its validity (step 2401) and extracts a hash value 2408 embedded in the extracted mark 2407 as a digital watermark (step 2402). The terminal 1101 also calculates a hash value 2409 of the Web page data except the part related to the mark whose validity is to be checked (step 2403) and compares the calculated hash value 2409 with the hash value 2408 extracted from the mark (step 2404). If they match, the terminal 1101 displays a message stating that the mark was validated on the display unit 1102; if they do not match, the terminal 1101 displays a message stating that the mark was not validated on the display unit 1102 (step 2405).

Detailed Description Text (215):
That is, as shown in FIG. 29, the terminal 1800a first gets a public key 2910 of the mark management organization 1121 from the public key DB 1801. Then, the terminal 1800a extracts a mark 2908 from a Web page 2907 to check its validity (step 2901), extracts a digital signature 2909 embedded in the extracted mark 2908 as a digital watermark (step 2902), and decrypts the extracted digital signature using the public key 2910 of the mark management organization 1121 to get a hash value 2911 (step 2903). The terminal 1800a also calculates a hash value 2912 of the Web page data except the part related to the mark 2908 whose validity is to be checked (step 2904), and compares the calculated hash value 2912 with the hash value 2911 generated by decrypting the digital signature extracted from the mark 2908 (step 2905). If they match, the terminal 1800a displays a message on the display unit 1102 stating that the mark was validated; if they do not match, the terminal 1800a displays a message staging that the mark was not validated (step 2906).

Detailed Description Text (222):
That is, in the sixth embodiment, the consumer terminal extracts the mark to be validated from the Web page, and sends the extracted mark and a validity check request to the mark management server. In the seventh and eighth embodiments, the consumer terminal sends Web page data containing the mark and the validity check request to the mark management server. On the display unit of the consumer terminal there is displayed a successful or an unsuccessful validity check message sent back from the mark management server. On the other hand, upon receiving a validity check request, the mark management server performs the validity check on the mark in the same way as the consumer terminal performs in the sixth to eighth embodiments. In the sixth embodiment, the mark management server extracts information embedded in the mark sent with the request. If this information matches the information embedded by the mark management server, it sends a successful validity message to the consumer terminal; if not, it sends an unsuccessful validity check message to the consumer terminal. In the seventh embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the hash value embedded in the mark as the digital watermark, calculates the hash value of the Web page except the area related to the mark to be validated, and compares this value with the hash value extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal. In the eighth embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the digital signature embedded in the extracted mark as the digital watermark, and extracts the hash value by decrypting the digital signature with a public key of the mark management organization. The mark management server calculates the hash value of the Web page data except the area related to the mark to be validated, and compares this value with the hash value generated by decrypting the digital signature extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal.

L26: Entry 1 of 1 File: USPT Oct 10, 2000

DOCUMENT-IDENTIFIER: US 6131162 A
TITLE: Digital data authentication method

Application Filing Date (1):
19980604

Brief Summary Text (34):
Sometimes, a Web page may also contain image data, such as logo marks indicating the Web page creator or an authentic individual or organization which has authorized the Web page, to allow a Web page user to instantly ascertain who has created the Web page or that the Web page has been authorized by the authentic individual or organization.

Brief Summary Text (56):
To achieve the above object, this invention is an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of embedding a digital signature into the content such that the digital signature cannot be separated from the content without using a predetermined rule, the digital signature being created by encrypting an n-bit hash value using a private key in accordance with a public key cipher system used by a first content-handling person, the n-bit hash value being obtained by evaluating the content with a first hash function; and sequentially repeating digital signature embedding for a second person to a k-th content-handling person, wherein, for an i-th content-handling person (i is an integer between 2 and k), the content into which the digital signatures of the first to an (i-1) contenthandling persons are embedded is evaluated with a second hash function, wherein a resulting n/2-bit hash value is encrypted using the private key of the i-th content-handling person to generate the digital signature of the i-th content-handling person, and wherein the digital signature of the i-th content-handling person is embedded into the content in which the digital signatures from the first to the (i-1)th persons are already embedded such that the digital signature of the i-th content-handling person cannot be separated from the content without using a predetermined rule.

Brief Summary Text (58):
This invention is also an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

Detailed Description Text (14):
A program for creating the provider system 100 and the purchaser system 200 in an electronic computer system is loaded into the main storage 302 for execution by the CPU 301. The program is pre-recorded on the external storage unit 303b and is loaded, as necessary, into the main storage 302 for execution by the CPU 301.

Alternatively, the program is pre-recorded on a portable recording medium 307 such as a CD-ROM disc and is loaded directly, as necessary, via the external storage unit 303a for execution by the CPU 301. It is also possible that the program is installed from the portable recording medium 307 via the external storage unit 303a used for portable recording medium onto the external storage unit 303b such as a hard disk and is loaded, as necessary, into the main storage 302 for execution by the CPU 301.

Detailed Description Text (24):
Now, assume that the purchaser has created an illegal copy of the content which is stored in the storage module 220 and into which the digital signature is embedded (without an appropriate authority to create a copy) and has transferred the created copy to a third party. As explained in Description of Related Art, the purchaser cannot remove the digital signature, which is embedded in the content, for example, in the form of a digital watermark, from the content. That is, the purchaser cannot create a complete but illegal copy which has no digital signature embedded.

Detailed Description Text (27):
Next, the controlling module 112 tells the signature verifying module 114 to verify the digital signature (step 602). To do so, the signature verifying module 114 decrypts the extracted digital signature using the verification key 122 of a user stored in the storage module 120 and compares the resulting value with the hash value obtained by evaluating the original content in the storage module 120 with the use of the same one-way hash function as that used by the purchaser system 200. If the rule used by the purchaser system 200 to embed the digital signature into the content is known only to the provider and if the digital signature may be removed from the content according to that rule, the content from which the digital signature is removed may be used instead of the original content.

Detailed Description Text (199):
That is, as shown in FIG. 24, the terminal 1101 first extracts a mark 2407 from a Web page 2406 to check its validity (step 2401) and extracts a hash value 2408 embedded in the extracted mark 2407 as a digital watermark (step 2402). The terminal 1101 also calculates a hash value 2409 of the Web page data except the part related to the mark whose validity is to be checked (step 2403) and compares the calculated hash value 2409 with the hash value 2408 extracted from the mark (step 2404). If they match, the terminal 1101 displays a message stating that the mark was validated on the display unit 1102; if they do not match, the terminal 1101 displays a message stating that the mark was not validated on the display unit 1102 (step 2405).

Detailed Description Text (215):
That is, as shown in FIG. 29, the terminal 1800a first gets a public key 2910 of the mark management organization 1121 from the public key DB 1801. Then, the terminal 1800a extracts a mark 2908 from a Web page 2907 to check its validity (step 2901), extracts a digital signature 2909 embedded in the extracted mark 2908 as a digital watermark (step 2902), and decrypts the extracted digital signature using the public key 2910 of the mark management organization 1121 to get a hash value 2911 (step 2903). The terminal 1800a also calculates a hash value 2912 of the Web page data except the part related to the mark 2908 whose validity is to be checked (step 2904), and compares the calculated hash value 2912 with the hash value 2911 generated by decrypting the digital signature extracted from the mark 2908 (step 2905). If they match, the terminal 1800a displays a message on the display unit 1102 stating that the mark was validated; if they do not match, the terminal 1800a displays a message staging that the mark was not validated (step 2906).

Detailed Description Text (222):
That is, in the sixth embodiment, the consumer terminal extracts the mark to be validated from the Web page, and sends the extracted mark and a validity check

request to the mark management server. In the seventh and eighth embodiments, the consumer terminal sends Web page data containing the mark and the validity check request to the mark management server. On the display unit of the consumer terminal there is displayed a successful or an unsuccessful validity check message sent back from the mark management server. On the other hand, upon receiving a validity check request, the mark management server performs the validity check on the mark in the same way as the consumer terminal performs in the sixth to eighth embodiments. In the sixth embodiment, the mark management server extracts information embedded in the mark sent with the request. If this information matches the information embedded by the mark management server, it sends a successful validity message to the consumer terminal; if not, it sends an unsuccessful validity check message to the consumer terminal. In the seventh embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the hash value embedded in the mark as the digital watermark, calculates the hash value of the Web page except the area related to the mark to be validated, and compares this value with the hash value extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal. In the eighth embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the digital signature embedded in the extracted mark as the digital watermark, and extracts the hash value by decrypting the digital signature with a public key of the mark management organization. The mark management server calculates the hash value of the Web page data except the area related to the mark to be validated, and compares this value with the hash value generated by decrypting the digital signature extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal.

Detailed Description Text (225):
The programs used in each of the above-described embodiments may be recorded on various types of recording media, including a floppy disk, CD-ROM, DVD, and so forth for distribution to a unit on which they are executed. Alternatively, the programs may be downloaded to the unit from some other server connected to the network to which the unit is connected.

Current US Cross Reference Classification (2):
380/30

Current US Cross Reference Classification (3):
705/57

CLAIMS:

1. An embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer, the method comprising the steps of:

creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

sequentially repeating digital signature creation for a second person to a k-th content-handling person to create the digital signatures of the content-handling persons; and

embedding the digital signature of the k-th content-handling person into the content such that the digital signature of the k-th content-handling person cannot be separated from the content without using a predetermined rule, the digital

signature of the k-th content-handling person being obtained by performing said digital signature creation for the k-th content-handling person, wherein, during said digital signature creation processing for an i-th content-handling person (i is an integer between 2 and k), a value dependent on the digital signature of the (i-1)th

content-handling person is encrypted using the private key of the i-th content-handling person to generate the digital signature of the (i-)th content-handling person.

Previous Doc Next Doc Go to Doc#

L28: Entry 1 of 1 File: USPT Oct 10, 2000

DOCUMENT-IDENTIFIER: US 6131162 A
TITLE: Digital data authentication method

Brief Summary Text (56):
To achieve the above object, this invention is an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of embedding a digital signature into the content such that the digital signature cannot be separated from the content without using a predetermined rule, the digital signature being created by encrypting an n-bit hash value using a private key in accordance with a public key cipher system used by a first content-handling person, the n-bit hash value being obtained by evaluating the content with a first hash function; and sequentially repeating digital signature embedding for a second person to a k-th content-handling person, wherein, for an i-th content-handling person (i is an integer between 2 and k), the content into which the digital signatures of the first to an (i-1) contenthandling persons are embedded is evaluated with a second hash function, wherein a resulting n/2-bit hash value is encrypted using the private key of the i-th content-handling person to generate the digital signature of the i-th content-handling person, and wherein the digital signature of the i-th content-handling person is embedded into the content in which the digital signatures from the first to the (i-1)th persons are already embedded such that the digital signature of the i-th content-handling person cannot be separated from the content without using a predetermined rule.

Brief Summary Text (58):
This invention is also an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

CLAIMS:

1. An embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer, the method comprising the steps of:

creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

sequentially repeating digital signature creation for a second person to a k-th content-handling person to create the digital signatures of the content-handling persons; and

embedding the digital signature of the k-th content-handling person into the content such that the digital signature of the k-th content-handling person cannot be separated from the content without using a predetermined rule, the digital signature of the k-th content-handling person being obtained by performing said digital signature creation for the k-th content-handling person, wherein, during said digital signature creation processing for an i-th content-handling person (i is an integer between 2 and k), a value dependent on the digital signature of the (i-1)th

content-handling person is encrypted using the private key of the i-th content-handling person to generate the digital signature of the (i-)th content-handling person.

Previous Doc Next Doc Go to Doc#

L24: Entry 4 of 4 File: USPT Sep 22, 1998

DOCUMENT-IDENTIFIER: US 5812669 A
TITLE: Method and system for providing secure EDI over an open network

Application Filing Date (1):
19950719

Brief Summary Text (9):
The ability to trade or conduct business on a peer-to-peer basis over an open public network, such as the INTERNET, without the need for password management may be controlled, to the extent desired, by the trading participants through the use of trading partner agreements to provide key exchange certification, or by reliance on a certificate authority which issues and verifies public/private key paths. Thus, private and secure transactions, subject to authentication and non-repudiation of both origin and receipt, along with verification of message integrity, using EDI, may be conducted over an open communication network.

Current US Cross Reference Classification (1):
380/30

Current US Cross Reference Classification (2):
705/75

CLAIMS:

1. In a public key/private key secure communication system for selectively interconnecting a plurality of computers over an open public network, said plurality of computers comprising a sender computer and a recipient computer, said sender and recipient computers exchanging secure digital messages there between, said sender computer having a first associated public key and a first associated private key, said recipient computer having a second associated public key and a second associated private key, said digital messages comprising an EDI interchange communication between said sender computer and said recipient computer, said EDI interchange communication having an associated EDI acknowledgment message; the improvement in said secure open network communication system comprising

means for computing a first hash for said EDI interchange communication from said sender computer;

means for inserting said first hash in a predetermined location in said associated EDI acknowledgment message;

means for computing a second hash of said associated EDI acknowledgment message;

means for digitally signing said associated EDI acknowledgment message, said message digitally signing means comprising means for encrypting said second hash with said sender computer's private key;

means for inserting said second hash in a predetermined location in said associated

EDI acknowledgment message;

means for transmitting said EDI interchange communication along with said digitally signed associated EDI acknowledgment message to said recipient computer over said open public network; and

means associated with said recipient computer for receiving and processing said received EDI interchange communication and said digitally signed EDI acknowledgment message for providing authentication and non-repudiation of said EDI interchange communication from said sender computer, said means comprising means for decrypting said encrypted second hash with said sender computer's public key; whereby secure private EDI interchange communications can occur over said open public network while providing authentication and non-repudiation of said EDI communications.

2. An improved secure open network communication system in accordance with claim 1 wherein said means associated with said recipient computer further comprises means for computing a third hash of said received EDI acknowledgement message; and means for comparing said third hash with said decrypted second hash from said received EDI acknowledgement message, said comparing means comprising means for providing an indication of integrity of said EDI acknowledgement message and non-repudiation of origin when said decrypted second hash and said third hash match.

3. An improved secure open network communication system in accordance with claim 2 wherein said means associated with said recipient computer further comprises means for computing a fourth hash of said received EDI interchange communication; and means for comparing said fourth hash of said received EDI interchange communication with said first hash in said received EDI acknowledgement message, said comparing means comprising means for providing an indication of integrity and verification of authenticity of said EDI interchange communication and non-repudiation of origin when said first and fourth hash match.

5. An improved secure open network communication system in accordance with claim 4 further comprising means associated with said sender computer for receiving said transmitted reply EDI acknowledgement message, and for decrypting said encrypted fifth hash with said recipient computer's public key for verifying said digital signature of said reply EDI acknowledgement message; and means for computing a sixth hash of said received reply reply EDI acknowledgement message; and means for comparing said sixth hash against said decrypted fifth hash, said comparing means comprising means for providing an indication of integrity of said received reply EDI acknowledgement message and non-repudiation of origin of said reply EDI acknowledgement message; whereby non-repudiation of receipt of said EDI interchange communication is established by said sender computer.

6. An improved secure open network communication system in accordance with claim 5 wherein said means for creating said reply EDI acknowledgement message further comprises means for inserting said fourth hash in a predetermined location in said transmitted reply EDI acknowledgement message, and said means associated with said sender computer further comprises means for comparing said fourth hash in said received reply EDI acknowledgement message with said first hash, said comparing means providing an indication of integrity and authenticity of said EDI interchange when said first and fourth hash match.

12. An improved secure open network communication system in accordance with claim 4 wherein said means for creating said reply EDI acknowledgement message further comprises means for inserting said fourth hash in a predetermined location in said transmitted reply EDI acknowledgement message, and said means associated with said sender computer further comprises means for comparing said fourth hash in said received reply EDI acknowledgement message with said first hash, said comparing means providing an indication of integrity and authenticity of said EDI interchange when said first and fourth hash match.

16. An improved secure open network communication system in accordance with claim 15 further comprising means associated with said sender computer for receiving said transmitted reply EDI acknowledgement message, and for decrypting said encrypted third hash with said recipient computer's public key for verifying said digital signature of said reply EDI acknowledgement message; and means for computing a fourth hash of said received reply reply EDI acknowledgement message; and means for comparing said fourth hash against said decrypted third hash, said comparing means comprising means for providing an indication of integrity of said received reply EDI acknowledgement message and non-repudiation of origin of said reply EDI acknowledgement message; whereby non-repudiation of receipt of said EDI interchange communication is established by said sender computer.

17. An improved secure open network communication system in accordance with claim 14 further comprising means associated with said sender computer for receiving said transmitted reply EDI acknowledgement message, and for decrypting said encrypted third hash with said recipient computer's public key for verifying said digital signature of said reply EDI acknowledgement message; and means for computing a fourth hash of said received reply reply EDI acknowledgement message; and means for comparing said fourth hash against said decrypted third hash, said comparing means comprising means for providing an indication of integrity of said received reply EDI acknowledgement message and non-repudiation of origin of said reply EDI acknowledgement message; whereby non-repudiation of receipt of said EDI interchange communication is established by said sender computer.

Previous Doc Next Doc Go to Doc#

L24: Entry 2 of 4 File: USPT Oct 10, 2000

DOCUMENT-IDENTIFIER: US 6131162 A
TITLE: Digital data authentication method

Application Filing Date (1):
19980604

Brief Summary Text (34):
Sometimes, a Web page may also contain image data, such as logo marks indicating the Web page creator or an authentic individual or organization which has authorized the Web page, to allow a Web page user to instantly ascertain who has created the Web page or that the Web page has been authorized by the authentic individual or organization.

Brief Summary Text (56):
To achieve the above object, this invention is an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of embedding a digital signature into the content such that the digital signature cannot be separated from the content without using a predetermined rule, the digital signature being created by encrypting an n-bit hash value using a private key in accordance with a public key cipher system used by a first content-handling person, the n-bit hash value being obtained by evaluating the content with a first hash function; and sequentially repeating digital signature embedding for a second person to a k-th content-handling person, wherein, for an i-th content-handling person (i is an integer between 2 and k), the content into which the digital signatures of the first to an (i-1) contenthandling persons are embedded is evaluated with a second hash function, wherein a resulting n/2-bit hash value is encrypted using the private key of the i-th content-handling person to generate the digital signature of the i-th content-handling person, and wherein the digital signature of the i-th content-handling person is embedded into the content in which the digital signatures from the first to the (i-1)th persons are already embedded such that the digital signature of the i-th content-handling person cannot be separated from the content without using a predetermined rule.

Brief Summary Text (58):
This invention is also an embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer. The method includes the steps of creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

Detailed Description Text (24):
Now, assume that the purchaser has created an illegal copy of the content which is stored in the storage module 220 and into which the digital signature is embedded (without an appropriate authority to create a copy) and has transferred the created copy to a third party. As explained in Description of Related Art, the purchaser cannot remove the digital signature, which is embedded in the content, for example, in the form of a digital watermark, from the content. That is, the purchaser cannot

create a complete but illegal copy which has no digital signature embedded.

Detailed Description Text (27):
Next, the controlling module 112 tells the signature verifying module 114 to verify the digital signature (step 602). To do so, the signature verifying module 114 decrypts the extracted digital signature using the verification key 122 of a user stored in the storage module 120 and compares the resulting value with the hash value obtained by evaluating the original content in the storage module 120 with the use of the same one-way hash function as that used by the purchaser system 200. If the rule used by the purchaser system 200 to embed the digital signature into the content is known only to the provider and if the digital signature may be removed from the content according to that rule, the content from which the digital signature is removed may be used instead of the original content.

Detailed Description Text (199):
That is, as shown in FIG. 24, the terminal 1101 first extracts a mark 2407 from a Web page 2406 to check its validity (step 2401) and extracts a hash value 2408 embedded in the extracted mark 2407 as a digital watermark (step 2402). The terminal 1101 also calculates a hash value 2409 of the Web page data except the part related to the mark whose validity is to be checked (step 2403) and compares the calculated hash value 2409 with the hash value 2408 extracted from the mark (step 2404). If they match, the terminal 1101 displays a message stating that the mark was validated on the display unit 1102; if they do not match, the terminal 1101 displays a message stating that the mark was not validated on the display unit 1102 (step 2405).

Detailed Description Text (215):
That is, as shown in FIG. 29, the terminal 1800a first gets a public key 2910 of the mark management organization 1121 from the public key DB 1801. Then, the terminal 1800a extracts a mark 2908 from a Web page 2907 to check its validity (step 2901), extracts a digital signature 2909 embedded in the extracted mark 2908 as a digital watermark (step 2902), and decrypts the extracted digital signature using the public key 2910 of the mark management organization 1121 to get a hash value 2911 (step 2903). The terminal 1800a also calculates a hash value 2912 of the Web page data except the part related to the mark 2908 whose validity is to be checked (step 2904), and compares the calculated hash value 2912 with the hash value 2911 generated by decrypting the digital signature extracted from the mark 2908 (step 2905). If they match, the terminal 1800a displays a message on the display unit 1102 stating that the mark was validated; if they do not match, the terminal 1800a displays a message staging that the mark was not validated (step 2906).

Detailed Description Text (222):
That is, in the sixth embodiment, the consumer terminal extracts the mark to be validated from the Web page, and sends the extracted mark and a validity check request to the mark management server. In the seventh and eighth embodiments, the consumer terminal sends Web page data containing the mark and the validity check request to the mark management server. On the display unit of the consumer terminal there is displayed a successful or an unsuccessful validity check message sent back from the mark management server. On the other hand, upon receiving a validity check request, the mark management server performs the validity check on the mark in the same way as the consumer terminal performs in the sixth to eighth embodiments. In the sixth embodiment, the mark management server extracts information embedded in the mark sent with the request. If this information matches the information embedded by the mark management server, it sends a successful validity message to the consumer terminal; if not, it sends an unsuccessful validity check message to the consumer terminal. In the seventh embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the hash value embedded in the mark as the digital watermark, calculates the hash value of the Web page except the area related to the mark to be validated, and compares this value

with the hash value extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal. In the eighth embodiment, the mark management server extracts the mark from the Web page sent with the request, extracts the digital signature embedded in the extracted mark as the digital watermark, and extracts the hash value by decrypting the digital signature with a public key of the mark management organization. The mark management server calculates the hash value of the Web page data except the area related to the mark to be validated, and compares this value with the hash value generated by decrypting the digital signature extracted from the mark. If they match, the mark management server sends a successful validity check message to the consumer terminal, and if not, it sends an unsuccessful message to the consumer terminal.

Current US Cross Reference Classification (2):
380/30

Current US Cross Reference Classification (3):
705/57

CLAIMS:

1. An embed-in-content information processing method for embedding information on k (k is an integer equal to or larger than 2) content-handling persons using an electronic computer, the method comprising the steps of:

creating a digital signature of a first content-handling person by encrypting a hash value using a private key in accordance with a public key cipher system of the first content-handling person, the hash value being created by evaluating the content with a first hash function;

sequentially repeating digital signature creation for a second person to a k-th content-handling person to create the digital signatures of the content-handling persons; and

embedding the digital signature of the k-th content-handling person into the content such that the digital signature of the k-th content-handling person cannot be separated from the content without using a predetermined rule, the digital signature of the k-th content-handling person being obtained by performing said digital signature creation for the k-th content-handling person, wherein, during said digital signature creation processing for an i-th content-handling person (i is an integer between 2 and k), a value dependent on the digital signature of the (i-1)th

content-handling person is encrypted using the private key of the i-th content-handling person to generate the digital signature of the (i-)th content-handling person.

Previous Doc Next Doc Go to Doc#

L24: Entry 1 of 4 File: USPT Mar 6, 2001

DOCUMENT-IDENTIFIER: US 6199053 B1
TITLE: Digital signature purpose encoding

Abstract Text (1):
A method and apparatus for encoding a purpose into a digital signature, where purpose and digital signature bound into an extended digital signature. The extended digital signature capability binds a purpose description identifying the purpose for the digital signature so that when affixed to a digital signature, the digital signature cannot be employed for improper purposes. A hash function is used to generate a hash value from the purpose description. The hash value is used in a digital signature function to bind the purpose to a digital signature. The extended digital signature can be verified for validity by comparing it to a hash value. In an electronic transaction, the extended digital signature can allow a purpose to be bound with the digital signature so that improper or unauthorized transactions are detected and disallowed.

Application Filing Date (1):
19990408

Brief Summary Text (8):
For many electronic commerce and digital content distribution applications, there is a requirement not only to generate and verify digital signatures, but also to control and enforce the purpose for which a digital signature was generated and verified. For instance, in an electronic commerce transaction, when a bank "signs" a purchase order of a credit card holder that it serves and submits it to the merchant, the bank may want to limit its authorization of the credit card to a set amount to prevent any overcharging of the account.

Brief Summary Text (11):
A method and apparatus for encoding a purpose description into the generation of digital signatures. Purpose description encoding allows an entity during an electronic transaction to control and enforce the purpose authorized for a digital signature.

Brief Summary Text (12):
An input data stream is passed through a hash function and utilized to generate a first hash value. The first hash value is used as a seed in generating an extended digital signature which amalgamates the purpose description into an ordinary digital signature. A purpose description is passed through a hash function (computer program/code which executes hashing), seeding with the first hash value to generate an extended hash value. The extended hash value is passed to a digital signature function (computer program/code which generates a digital signature) to generate an extended digital signature. The digital signature function can generate a digital signature which may be affixed to a transaction or document. When the extended digital signature is provided along with the purpose description to another digital signature function, generated by a different process or computer, both digital signature and purpose description can be decrypted to verify their authenticity. The extended digital signature capability and/or verification can be implemented as a service for function calls from other computers or processes, each function call passing its own digital signatures and purpose descriptions.

Detailed Description Text (7):
FIG. 2 is a diagram of ordinary digital signature verification without purpose binding. The first step in verifying a digital signature is to pass the data 100 to be verified through a hash function 200 to generate new hash value 310. Initialization block 150 operates identically as described with respect to FIG. 1 in providing an initial seed for the first segment of data and using intermediary hash values in succeeding segments. New hash value 310 and digital signature 500 are passed to the digital signature function 400. Digital signature function 400 decrypts digital signature 500 with the public key of the originator and recovers hash value 300 of FIG. 1. Hash value 300 is then compared to new hash value 310. If hash value 300 matches new hash value 310, then digital signature 500 is valid (510). If hash value 300 does not match new hash value 310, then the digital signature 500 is invalid (520).

Detailed Description Text (16):
Digital signature function 4400 decrypts digital signature 5001 with the public key of the originator and recovers extended hash value 4300. Extended hash value 4300 is then compared to extended hash value 4310. If extended hash value 4300 matches extended hash value 4310, the digital signature 5001 is valid (5100). If extended hash value 4300 does not match extended hash value 4310, the digital signature 5001 is invalid (5200).

Detailed Description Text (22):
The input digital signature and hash value of each function call, Function A 315, Function B 325, and Function C 335, are separately passed through the extended digital signature function to verify three separate digital signatures. For instance, if Function A 315 calls upon the EDSC to verify a digital signature, then purpose description 4100 would equal Purpose Description A 4110, and digital signature 5001 would equal digital signature 5010, and new hash value 3100 would equal new hash value 310. Thus the digital signature verified would be specific to Function A 315 and include purpose description A 4110 within extended hash value 4310. Extended hash value 4310 would then be compared with the extended hash value 4300 recovered by digital signature function 4400 for verification (5100--valid and 5200--invalid).

Detailed Description Text (25):
Server 10 is shown capable of receiving and sending data over some form of network or communications interface 50 (such as LAN or the Internet) to a client 5. Client 5, in an electronic commerce setting, may be a purchaser/customer attempting to transact with a merchant 15. Merchant 15 is also connected over network 50 to receive from and send data to server 10, client 5 and also a server 20. In the electronic commerce context, server 20 may be a service which verifies signatures submitted to them while server 10 may be the bank issuing credit or authorizing release of the funds of a purchaser held in an account at that bank.

Detailed Description Text (26):
If client 5 wishes to purchase or order a product or service from merchant 15, it would submit to its bank, server 10, purchasing data 25 identifying the product/service desired, the account number of the purchaser with the bank and the price the purchaser should pay. This purchasing data 25 may be sent over some secure transmission protocol so that the data relating to account information cannot be intercepted and/or misappropriated while being transmitted over network 50. Server 10 receives purchasing data 25 and affixes a signature to the data along with a purpose description which limits or qualifies the authorization represented by the signature.

Detailed Description Text (27):
In one embodiment, the digital signature is generated by server 10 itself and then forwarded directly to merchant 15 for processing the order. In another embodiment,

the signature or authorization and purpose may be wrapped as a function call to an application program interface on client 5, and therefore the purpose 35 (with or without data 25) is sent back to client 5. In this embodiment, an application program interface of client 5 provides an extended digital signature functionality in hardware or software which will bind the purpose 35 with the digital signature amalgamating them with purchasing data 25.

Detailed Description Text (35):
Digital signature processor 730 is coupled to or contains two circuits--hash function circuit 733 and digital signature circuit 735. Hash function circuit 733 receives a data stream over bus 740 from main memory 710 (or other component) which it uses along with a pre-specified initialization block to hash the data stream by segmenting it in main memory 710 as needed to produce a first hash value. Hash function circuit 733 cooperates with CPU 700 to perform this computation to arrive at the hash value. The hash value is fed back into hash function circuit 733 along with a purpose description 750 to compute a new, extended hash value. The extended hash value is passed to the digital signature function circuit to produce an extended digital signature. Though the digital signature processor is shown as a separate processor residing off the bus, it may be implemented as an embedded processor within the CPU or as plug-in module (e.g., SRAM or Flash).

Current US Original Classification (1):
705/76

Current US Cross Reference Classification (2):
380/30

CLAIMS:

4. The method of claim 1, wherein the purpose description includes information to limit a scope of authority represented by the extended digital signature.

5. The method of claim 1, wherein the purpose description includes information to qualify a scope of authority represented by the extended digital signature.

7. The method of claim 1 wherein producing an extended hash value comprises seeding a first hash function with a hash value generated from the input data, passing the purpose description to the hash function, and generating the extended hash value from the purpose description and the first hash value.

8. The method of claim 1 wherein producing an extended hash value comprises passing an initialization block and the purpose description to a first hash function, the first hash function generating a first extended hash value, and generating the extended hash value from the first hash value.

9. The method of claim 8 wherein producing an extended hash value comprises passing the data stream to a second hash function generating an initialization hash value, the initialization hash value being used as the initialization block.

13. The apparatus of claim 10, wherein the purpose description includes information to limit a scope of authority represented by the extended digital signature.

14. The apparatus of claim 10, wherein the purpose description includes information to qualify a scope of authority represented by the extended digital signature.

18. The apparatus of claim 10 wherein the hash function circuit is operative to produce the hash value by seeding a first hash function with a hash value generated from the input data signals, passing the purpose description to the hash function, and generating the hash value from the purpose description and the first hash value.

19. The apparatus of claim 10 wherein the hash function circuit is operative to produce the hash value by passing an initialization block and the purpose description to a first hash function, the first hash function generating a first extended hash value, and generating the hash value from the first hash value.

20. The apparatus of claim 19 wherein the hash function circuit is further operative to produce the hash value by passing the input data signals to a second hash function generating an initialization hash value, the initialization hash value being used as the initialization block.

24. The medium of claim 21, wherein the purpose description includes information to limit a scope of authority represented by the extended digital signature.

25. The medium of claim 21, wherein the purpose description includes information to qualify a scope of authority represented by the extended digital signature.

26. The medium of claim 25 wherein the instructions for producing an extended hash value further comprise instructions causing the processor to perform operations comprising passing the data stream to a second hash function generating an initialization hash value, the initialization hash value being used as the initialization block.

28. The medium of claim 21 wherein the instructions for producing an extended hash value further comprise instructions causing the processor to perform operations comprising seeding a first hash function with a hash value generated from the input data, passing the purpose description to the hash function, and generating the extended hash value from the purpose description and the first hash value.

29. The medium of claim 21 wherein the instructions for producing an extended hash value further comprise instructions causing the processor to perform operations comprising passing an initialization block and the purpose description to a first hash function, the first hash function generating a first extended hash value, and generating the extended hash value from the first hash value.

33. The apparatus of claim 30, wherein the purpose description includes information to limit scope of authority represented by the extended digital signature.

34. The apparatus of claim 30, wherein the purpose description includes information to qualify a scope of authority represented by the extended digital signature.

36. The apparatus of claim 30 wherein the means for producing an extended hash value includes means for seeding a first hash function with a hash value generated from the input data, means for passing the purpose description to the hash function, and means for generating the extended hash value from the purpose description and the first hash value.

37. The apparatus of claim 30, wherein the means for producing an extended hash value includes means for passing an initialization block and the purpose description to a first hash function, the first hash function generating a first extended hash value, and means for generating the extended hash value from the first hash value.

38. The apparatus of claim 30 wherein the means for producing an extended hash value, include means for passing the data stream to a second hash function generating an initialization hash value, the initialization hash value being used as the initialization block.

39. A machine-readable medium having stored thereon data representing sequences of instructions which, when executed by a processor, cause the processor to perform

operations comprising:

passing a purpose description and a digital signature to an extended digital signature function;

generating a first extended hash value using the purpose description;

recovering a second extended hash value from the digital signature; and

comparing the first extended hash value with the second extended hash value, a true comparison resulting in verifying that the digital signature and the purpose description are valid.

40. The medium of claim 39 wherein the instructions for generating the first extended hash value further comprise instructions causing the machine to perform operations comprising passing an initialization block and the purpose description to a first hash function, the first hash function generating the first extended hash value.

41. The medium of claim 39 wherein the instructions further comprise instructions causing the machine to perform operations comprising passing the data stream to a second hash function generating an initialization hash value, the initialization hash value used as the initialization block.

45. An apparatus for verifying a digital signature and a purpose description for a data stream comprising:

means for passing the purpose description and the digital signature to an extended digital signature function;

means for generating a first extended hash value using the purpose description;

means for recovering a second extended hash value from the digital signature; and

means for comparing the first extended hash value with the second extended hash value, a true comparison resulting in verifying that the digital signature and the purpose description are valid.

46. The apparatus of claim 45 wherein the means for generating the first extended hash value includes means for passing an initialization block and the purpose description to a first hash function, the first hash function generating the first extended hash value.

47. The apparatus of claim 45 further comprising means for passing the data stream to a second hash function generating an initialization hash value, the initialization hash value used as the initialization block.

Previous Doc Next Doc Go to Doc#

L24: Entry 3 of 4 File: USPT Feb 8, 2000

DOCUMENT-IDENTIFIER: US 6023509 A
TITLE: Digital signature purpose encoding

Abstract Text (1):
A method and apparatus for encoding a purpose into a digital signature, where purpose and digital signature bound into an extended digital signature. The extended digital signature capability binds a purpose description identifying the purpose for the digital signature so that when affixed to a digital signature, the digital signature cannot be employed for improper purposes. A hash function is used to generate a hash value from the purpose description. The hash value is used in a digital signature function to bind the purpose to a digital signature. The extended digital signature can be verified for validity by comparing it to a hash value. In an electronic transaction, the extended digital signature can allow a purpose to be bound with the digital signature so that improper or unauthorized transactions are detected and disallowed.

Application Filing Date (1):
19960930

Brief Summary Text (8):
For many electronic commerce and digital content distribution applications, there is a requirement not only to generate and verify digital signatures, but also to control and enforce the purpose for which a digital signature was generated and verified. For instance, in an electronic commerce transaction, when a bank "signs" a purchase order of a credit card holder that it serves and submits it to the merchant, the bank may want to limit its authorization of the credit card to a set amount to prevent any overcharging of the account.

Brief Summary Text (11):
The invention is a method and apparatus for encoding a purpose description into a digital signature. Purpose description encoding allows an entity during an electronic transaction to control and enforce the purpose authorized for a digital signature.

Brief Summary Text (12):
An input data stream is passed through a hash function and utilized to generate a first hash value. The first hash value is used as a seed in generating an extended digital signature which amalgamates the purpose description into an ordinary digital signature. A purpose description is passed through a hash function (computer program/code which executes hashing), seeding with the first hash value to generate an extended hash value. The extended hash value is passed to a digital signature function (computer program/code which generates a digital signature) to generate an extended digital signature. The digital signature function can generate a digital signature which may be affixed to a transaction or document. When the extended digital signature is provided along with the purpose description to another digital signature function, generated by a different process or computer, both digital signature and purpose description can be decrypted to verify their authenticity. The extended digital signature capability and/or verification can be implemented as a service for function calls from other computers or processes, each function call passing its own digital signatures and purpose descriptions.

Detailed Description Text (6):
FIG. 2 is a diagram of ordinary digital signature verification without purpose binding. The first step in verifying a digital signature is to pass the data 100 to be verified through a hash function 200 to generate new hash value 310. Initialization block 150 operates identically as described with respect to FIG. 1 in providing an initial seed for the first segment of data and using intermediary hash values in succeeding segments. New hash value 310 and digital signature 500 are passed to the digital signature function 400. Digital signature function 400 decrypts digital signature 500 with the public key of the originator and recovers hash value 300 of FIG. 1. Hash value 300 is then compared to new hash value 310. If hash value 300 matches new hash value 310, then digital signature 500 is valid (510). If hash value 300 does not match new hash value 310, then the digital signature 500 is invalid (520).

Detailed Description Text (15):
Digital signature function 4400 decrypts digital signature 5001 with the public key of the originator and recovers extended hash value 4300. Extended hash value 4300 is then compared to extended hash value 4310. If extended hash value 4300 matches extended hash value 4310, the digital signature 5001 is valid (5100). If extended hash value 4300 does not match extended hash value 4310, the digital signature 5001 is invalid (5200).

Detailed Description Text (21):
The input digital signature and hash value of each function call, Function A 315, Function B 325, and Function C 335, are separately passed through the extended digital signature function to verify three separate digital signatures. For instance, if Function A 315 calls upon the EDSC to verify a digital signature, then purpose description 4100 would equal Purpose Description A 4110, and digital signature 5001 would equal digital signature 5010, and new hash value 3100 would equal new hash value 310. Thus the digital signature verified would be specific to Function A 315 and include purpose description A 4110 within extended hash value 4310. Extended hash value 4310 would then be compared with the extended hash value 4300 recovered by digital signature function 4400 for verification (5100--valid and 5200--invalid).

Detailed Description Text (24):
Server 10 is shown capable of receiving and sending data over some form of network or communications interface 50 (such as LAN or the Internet) to a client 5. Client 5, in an electronic commerce setting, may be a purchaser/customer attempting to transact with a merchant 15. Merchant 15 is also connected over network 50 to receive from and send data to server 10, client 5 and also a server 20. In the electronic commerce context, server 20 may be a service which verifies signatures submitted to them while server 10 may be the bank issuing credit or authorizing release of the funds of a purchaser held in an account at that bank.

Detailed Description Text (25):
If client 5 wishes to purchase or order a product or service from merchant 15, it would submit to its bank, server 10, purchasing data 25 identifying the product/service desired, the account number of the purchaser with the bank and the price the purchaser should pay. This purchasing data 25 may be sent over some secure transmission protocol so that the data relating to account information cannot be intercepted and/or misappropriated while being transmitted over network 50. Server 10 receives purchasing data 25 and affixes a signature to the data along with a purpose description which limits or qualifies the authorization represented by the signature.

Detailed Description Text (26):
In one embodiment, the digital signature is generated by server 10 itself and then forwarded directly to merchant 15 for processing the order. In another embodiment,

the signature or authorization and purpose may be wrapped as a function call to an application program interface on client 5, and therefore the purpose 35 (with or without data 25) is sent back to client 5. In this embodiment, an application program interface of client 5 provides an extended digital signature functionality in hardware or software which will bind the purpose 35 with the digital signature amalgamating them with purchasing data 25.

Detailed Description Text (34):
Digital signature processor 730 is coupled to or contains two circuits--hash function circuit 733 and digital signature circuit 735. Hash function circuit 733 receives a data stream over bus 740 from main memory 710 (or other component) which it uses along with a pre-specified initialization block to hash the data stream by segmenting it in main memory 710 as needed to produce a first hash value. Hash function circuit 733 cooperates with CPU 700 to perform this computation to arrive at the hash value. The hash value is fed back into hash function circuit 733 along with a purpose description 750 to compute a new, extended hash value. The extended hash value is passed to the digital signature function circuit to produce an extended digital signature. Though the digital signature processor is shown as a separate processor residing off the bus, it may be implemented as an embedded processor within the CPU or as plug-in module (e.g., SRAM or Flash).

Current US Original Classification (1):
705/76

Current US Cross Reference Classification (2):
380/30

CLAIMS:

1. A method for encoding a purpose description for a data stream comprising the steps of:

passing a purpose description to an extended digital signature function;

seeding a first hash function with a hash value generated from said data stream;

passing said purpose description to said hash function;

generating an extended hash value from said purpose description and said first hash value; and

generating an extended digital signature using a digital signature function on said extended hash value.

2. A method of verifying a digital signature and a purpose description for a data stream comprising the steps of:

passing said purpose description and said digital signature to an extended digital signature function;

generating a first extended hash value using said purpose description;

recovering a second extended hash value from said digital signature; and

comparing said first extended hash value with said second extended hash value, a true comparison resulting in verifying that digital signature and said purpose description are valid.

3. A method of verifying according to claim 2 further wherein the step of generating said first extended hash value includes the step of:

passing an initialization block and said purpose description to a first hash function, said first hash function generating said first extended hash value.

4. A method of verifying according to claim 3 further comprising the steps of:

passing said data stream to a second hash function generating an initialization hash value, said initialization hash value used as said initialization block.

10. In a network capable of handling an electronic transaction, an apparatus comprising:

a client coupled to said network requesting said electronic transaction, said client sending a request over said network;

a server coupled to said client, and to said network, said server receiving said request and sending an extended digital signature for authorizing said request, wherein a purpose description is binded in said extended digital signature; and

an electronic transaction provider, said provider coupled to said network, said provider completing said electronic transaction in accordance with said purpose description of said authorization.

13. A digital signature processor according to claim 12, further comprising a digital signature verification circuit, said digital signature verification circuit comparing said hash value with a second value, wherein if said comparison evaluates true, said digital signature is verified as valid.

Previous Doc Next Doc Go to Doc#

Hit List

Your wildcard search against 10000 terms has yielded the results below.
Your result set for the last L# is incomplete.
The probable cause is use of unlimited truncation. Revise your search strategy to use limited truncation.

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 10 of 10 returned.

☐ 1. Document ID: US 6772133 B1

L11: Entry 1 of 10 File: USPT Aug 3, 2004

US-PAT-NO: 6772133
DOCUMENT-IDENTIFIER: US 6772133 B1

TITLE: Information recording device and information reproducting device

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 2. Document ID: US 6526146 B1

L11: Entry 2 of 10 File: USPT Feb 25, 2003

US-PAT-NO: 6526146
DOCUMENT-IDENTIFIER: US 6526146 B1

TITLE: Information recording system

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 3. Document ID: US 6367013 B1

L11: Entry 3 of 10 File: USPT Apr 2, 2002

US-PAT-NO: 6367013
DOCUMENT-IDENTIFIER: US 6367013 B1
** See image for Certificate of Correction **

TITLE: System and method for electronic transmission, storage, and retrieval of authenticated electronic original documents

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☑ 4. Document ID: US 6233684 B1

h e b b g e e e f e h ef b e

L11: Entry 4 of 10 File: USPT May 15, 2001

US-PAT-NO: 6233684
DOCUMENT-IDENTIFIER: US 6233684 B1
** See image for Certificate of Correction **

TITLE: System for controlling the distribution and use of rendered digital works through watermaking

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 5. Document ID: US 6141754 A

L11: Entry 5 of 10 File: USPT Oct 31, 2000

US-PAT-NO: 6141754
DOCUMENT-IDENTIFIER: US 6141754 A

TITLE: Integrated method and system for controlling information access and distribution

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 6. Document ID: US 6141753 A

L11: Entry 6 of 10 File: USPT Oct 31, 2000

US-PAT-NO: 6141753
DOCUMENT-IDENTIFIER: US 6141753 A

TITLE: Secure distribution of digital representations

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 7. Document ID: US 6073123 A

L11: Entry 7 of 10 File: USPT Jun 6, 2000

US-PAT-NO: 6073123
DOCUMENT-IDENTIFIER: US 6073123 A

TITLE: Method and apparatus for detecting unauthorized copies of software

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 8. Document ID: US 6002768 A

L11: Entry 8 of 10 File: USPT Dec 14, 1999

h e b b g e e e f e h ef b e

US-PAT-NO: 6002768
DOCUMENT-IDENTIFIER: US 6002768 A

TITLE: Distributed registration and key distribution system and method

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 9. Document ID: US 5970143 A

L11: Entry 9 of 10 File: USPT Oct 19, 1999

US-PAT-NO: 5970143
DOCUMENT-IDENTIFIER: US 5970143 A

TITLE: Remote-auditing of computer generated outcomes, authenticated billing and access control, and software metering system using cryptographic and other protocols

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 10. Document ID: US 5715403 A

L11: Entry 10 of 10 File: USPT Feb 3, 1998

US-PAT-NO: 5715403
DOCUMENT-IDENTIFIER: US 5715403 A

TITLE: System for controlling the distribution and use of digital works having attached usage rights where the usage rights are defined by a usage rights grammar

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L6 and ((another or additon$ or new or extra) near2 cop$)	10

Display Format: TI Change Format

Previous Page Next Page Go to Doc#

h e b b g e e e f e h ef b e

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

Generate Collection Print

L11: Entry 5 of 10 File: USPT Oct 31, 2000

US-PAT-NO: 6141754
DOCUMENT-IDENTIFIER: US 6141754 A

TITLE: Integrated method and system for controlling information access and distribution

DATE-ISSUED: October 31, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Choy; David M.	Los Altos	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Business Machines Corporation	Armonk	NY			02

APPL-NO: 08/ 979713 [PALM]
DATE FILED: November 28, 1997

INT-CL: [07] G06 F 17/30, G06 F 12/14

US-CL-ISSUED: 713/200; 705/52, 705/59
US-CL-CURRENT: 713/200; 705/52, 705/59

FIELD-OF-SEARCH: 713/200, 713/201, 713/202, 713/189, 713/194, 709/229, 709/226, 711/163, 705/51, 705/52, 705/55, 705/57, 705/59

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5295266	March 1994	Hinsley et al.	709/101
☐	5629980	May 1997	Stefik et al.	380/4
☐	5634012	May 1997	Stefik et al.	705/39
☐	5638443	June 1997	Stefik et al.	705/54
☐	5649185	July 1997	Antognini et al.	
☐	5715403	February 1998	Stefik et al.	705/44
☐	5742759	April 1998	Nessett et al.	713/201

h e b b g e e e f c e h e ge

☐	5758068	May 1998	Brandt et al	713/200
☐	5758069	May 1998	Olsen	713/201
☐	5765152	June 1998	Erickson	707/1
☐	5826011	October 1998	Chou et al.	713/200
☐	5893910	April 1999	Martineau et al.	707/10
☐	5922073	July 1999	Shimada	713/200
☐	5922074	July 1999	Richard et al.	713/200
☐	5941947	August 1999	Brown et al.	709/225
☐	6009525	December 1999	Horstmann	713/200
☐	6044469	March 2000	Horstmann	713/200

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
40-8263440	October 1996	JP	

ART-UNIT: 275

PRIMARY-EXAMINER: Beausoliel, Jr.; Robert W.

ASSISTANT-EXAMINER: Revak; Christopher

ATTY-AGENT-FIRM: Sughrue, Mion, Zinn, Macpeak & Seas, PLLC

ABSTRACT:

A distributed content entity includes a protection specification and an information entity, in which the protection specification and the information entity are attached and transported together. The protection specification includes information for controlling the use of the information entity. A framework generates the distributed content entity, in which the framework includes a protection specification unit storing the protection specification and including an access control enforcement manager and an enhanced access control enforcement manager; an information unit for storing the protected information entity; and an access checking unit connected to the protection specification unit and the information unit. The access checking unit checks whether a user has a privilege to access the protected information entity based on the protection specification and the access control manager, and checks whether the requested access meets conditions determined based on the protection specification and enforced by the enhanced access control manager. An example of the enhanced access control manager is a terms and conditions enforcement manager for enforcing the terms and conditions of an agreement relating to permitted uses of the protected information entity.

66 Claims, 6 Drawing figures

Previous Doc Next Doc Go to Doc#

h e b b g e e e f c e h e ge

L11: Entry 5 of 10 File: USPT Oct 31, 2000

DOCUMENT-IDENTIFIER: US 6141754 A
TITLE: Integrated method and system for controlling information access and distribution

Application Filing Date (1):
19971128

Brief Summary Text (12):
To provide such protection, the state of the current art is to use a conventional access control mechanism to protect information stored in a system and to provide a separate rights management mechanism to protect information distributed outside an administrative domain using encryption; content marking/finger-printing, digital signature, and other techniques. It should be noted that, while 100% protection of the rights of all parties involved in information distribution may not be feasible with current technologies, it is frequently not needed for many applications. For example, some "leakage" is often acceptable when protection is primarily on economic value rather than on secrecy, especially if a less-than-100% protection scheme allows the system to be more efficient and easier to use, and/or makes content more available, thereby increasing usage.

Brief Summary Text (13):
One design to protect distributed information, is for an information supplier to distribute information entities in encrypted form, such as with IBM Corporation's CRYPTOLOPE scheme as shown in FIG. 2 below. In this manner, information can be distributed freely using any means without loss of protection. A user who wants to use an encrypted entity must obtain the corresponding decryption key to "unlock" the content. This key can be obtained from a clearing center server 220, which can be the information supplier, an authorized agent of the supplier, or a mutually trusted third party who provides the clearing service. A clearing center 220 must verify that the user has satisfied the criteria to receive the entity according to the terms and conditions (T&Cs) associated with that entity (e.g., by paying a fee), before providing the user with the corresponding decryption key. Any business transaction (e.g., payment) can be handled by yet another party.

Brief Summary Text (27):
The invention is also directed to a method for transmitting a protected information entity, including generating a distributed content entity by combining the protected information entity with a protection specification, and transmitting the distributed content entity. The protection specification specifies access and privilege controls for using the protected information entity, and the method further includes encrypting the distributed content entity prior to transmission.

Drawing Description Text (4):
FIG. 2 is a block diagram showing a conventional system for distributing encrypted information;

Detailed Description Text (26):
The information usage meter 403 can be included to handle accounting and enforce usage-based T&Cs (e.g., a content subscription or lease that entitles a user to print a certain number of copies with extra printing at additional cost). The usage

meter is also used to produce various reports for authorized parties such as publishers and librarians.

Detailed Description Text (30):
When information content is to be distributed from one participant information system to another, or to a user, the information entity 302 and its corresponding protection specification 301 are retrieved via the access checking unit 401, through which the information entity is retrieved from the storage unit 440 and the corresponding protection specification is retrieved from the storage unit 430. The retrieved information entity and its corresponding protection specification are packaged into a distributed content entity 300. Suitable content distribution protection techniques (e.g., watermark, digital signature, encryption) can be applied to the distributed content entity 300 before it is distributed (not shown).

Detailed Description Text (31):
When a participant information system receives a distributed content entity 300, after decryption and authentication as needed, the information entity 302 and its protection specification 301 are extracted from the distributed content entity 300 and are stored into the system through the access checking unit 401. The information entity 302 is interpreted using the information model 420 and is stored in the protected information storage unit 440, whereas the protection specification is interpreted using the protection model 410 and is stored in the protection specification storage unit 430.

Detailed Description Text (43):
Optionally, the distributed content entity 300 can be encrypted by the participant information system 400. Upon receipt, the receiving user or another participant information system decrypts the received distributed content entity 300, as shown in FIG. 2. A decryption key can be obtained from either the other participant information system or the clearance center 220.

Detailed Description Text (54):
Optionally, the pushed distributed content entity 300 can be encrypted by the participant information system 400. Upon receipt, the receiving participant information control system decrypts the pushed distributed content entity 300, as shown in FIG. 2.

Current US Cross Reference Classification (2):
705/59

CLAIMS:

5. The distributed content entity according to claim 1, wherein the distributed content entity is encrypted.

26. The framework according to claim 12, wherein prior to sending the distributed content entity to the user, the distributed content entity is encrypted.

27. The framework according to claim 26, wherein the distributed content entity is encrypted according to an encryption method provided by a clearance center.

36. The apparatus according to claim 35, wherein prior to transmitting the distributed content entity the distributed content entity is encrypted.

37. The apparatus according to claim 36, wherein the distributed content entity is encrypted according to an encryption method provided by a clearance center.

46. The method according to claim 44, further comprising encrypting the distributed content entity prior to transmission.

h e b b g e e e f c e h e ge

62. The method according to claim 61, further comprising:

requesting a key for decrypting the information entity;

receiving the key only if authorization is granted based on the protection specification.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#
End of Result Set

Generate Collection Print

L14: Entry 3 of 3 File: USPT Aug 25, 1992

US-PAT-NO: 5142578
DOCUMENT-IDENTIFIER: US 5142578 A

TITLE: Hybrid public key algorithm/data encryption algorithm key distribution method based on control vectors

DATE-ISSUED: August 25, 1992

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Matyas; Stephen M.	Manassas	VA		
Johnson; Donald B.	Manassas	VA		
Le; An V.	Manassas	VA		
Prymak; Rostislaw	Dumfries	VA		
Wilkins; John D.	Somerville	VA		
Martin; William C.	Concord	NC		
Rohland; William S.	Charlotte	NC		

US-CL-CURRENT: 380/280; 380/281, 380/30, 713/175

ABSTRACT:

The patent describes a method and apparatus for securely distributing an initial Data Encryption Algorithm (DEA) key-encrypting key by encrypting a key record (consisting of the key-encrypting key and control information associated with that key-encrypting key) using a public key algorithm and a public key belonging to the intended recipient of the key record. The patent further describes a method and apparatus for securely recovering the distributed key-encrypting key by the recipient by decrypting the received key record using the same public key algorithm and private key associated with the public key and re-encrypting the key-encrypting key under a key formed by arithmetically combining the recipient's master key with a control vector contained in the control information of the received key record. Thus the type and usage attributes assigned by the originator of the key-encrypting key in the form of a control vector are cryptographically coupled to the key-encrypting key such that the recipient may only use the received key-encrypting key in a manner defined by the key originator.

The patent further describes a method and apparatus to improve the integrity of the key distribution process by applying a digital signature to the key record and by including identifying information (i.e., an originator identifier) in the control information of the key record. The integrity of the distribution process is enhanced by verifying the digital signature and originator identifier at the recipient node.

25 Claims, 16 Drawing figures
Exemplary Claim Number: 3
Number of Drawing Sheets: 9

h e b b g e e e f c e f e ge

L14: Entry 3 of 3 File: USPT Aug 25, 1992

DOCUMENT-IDENTIFIER: US 5142578 A
TITLE: Hybrid public key algorithm/data encryption algorithm key distribution method based on control vectors

Application Filing Date (1):
19910822

Brief Summary Text (16):
The cryptographic architecture described in the cited patents by S. M. Matyas, et al. is based on associating with a cryptographic key, a control vector which provides the authorization for the uses of the key intended by the originator of the key. The cryptographic architecture described in the cited patents by S. M. Matyas, et al. is based on the Data Encryption Algorithm (DEA), whereas the present invention is based on both a secret key algorithm, such as the DEA, and a public key algorithm. Various key management functions, data cryptography functions, and other data processing functions are possible using control vectors, in accordance with the invention. A system administrator can exercise flexibility in the implementation of his security policy by selecting appropriate control vectors in accordance with the invention. A cryptographic facility (CF) in the cryptographic architecture is described in the above cited patents by S. M. Matyas, et al. The CF is an instruction processor for a set of cryptographic instructions, implementing encryption methods and key generation methods. A memory in the crypto facility stores a set of internal cryptographic variables. Each cryptographic instruction is described in terms of a sequence of processing steps required to transform a set of input-parameters to a set of output parameters. A cryptographic facility application program is also described in the referenced patents and patent applications, which defines an invocation method, as a calling sequence, for each cryptographic instruction consisting of an instruction mnemonic and an address with corresponding input and output parameters.

Detailed Description Text (26):
FIG. 8 illustrates the scheme for DEA key distribution with digital signatures, which is the same as the scheme shown in FIG. 5 except as follows. Once the encrypted key value ePUb(keyblk) has been produced, the GKSP instruction additionally produces the digital signature DSIGa from ePUb(keyblk) and the private key PRa belonging to cryptographic system A. A common method for producing such a signature is to first calculate a hash value on ePUb(keyblk) using a one way cryptographic function, such as described in U.S. Pat. No. 4,908,861 by Brachtl et al., cited in the background art, which uses either two DEA encryptions or four DEA encryptions per each 64 bits of input text to bc hashed, and then decrypt (or transform) the hash value using the private key PRa to produce a DSIGa of the form dPRa(hash value). The clear value of PRa is obtained by decrypting the encrypted value of eKMa.C2(PRa) supplied as an input to the GKSP instruction at 313 using the DEA and the variant key KMa.C2. KMa.C2 is formed as the Exclusive OR product of master key KMa stored in CF 30 and control vector C2 supplied as input to the GKSP instruction at 313. For example, if the public key algorithm is the RSA algorithm, a the digital signature may be calculated using the method as described in ISO Draft International Standard 9796 entitled "Information Technology .SIGMA.Security

Techniques .SIGMA.Digital Signature Scheme Giving Message Recovery." The so-produced DSIGa 315 is returned as an output at 314. Both the external key token 308 and the DSIGa 315 are transmitted to cryptographic system B. At cryptographic system B, the IDK instruction is used to import the key K in similar fashion as described in FIG. 5 except that the IDK instruction first validates DSIGa using the public key PUa previously imported, encrypted, and stored in CKDS 22'. A DSIGa of the form dPRa(hash value) is validated by encrypting dPRa(hash value) with PUa, calculating a hash value on ePUb(keyblk) using the same one way cryptographic function, called the hash value of reference, and comparing the hash value of reference and the recovered clear hash value for equality. Only if this comparison check is successful does the IDK instruction continue and import the key K. The clear value of PUa is obtained by decrypting the encrypted value of eKMb.Cl(PUa) supplied as an input to the IDK instruction at 316 using the DEA and the variant key KMb.Cl. KMb.Cl is formed as the Exclusive OR product of master key KMb stored in CF 30' and control vector Cl supplied as input to the IDK instruction at 316. Thus, the GKSP instruction at cryptographic system A produces DSIGa and the IDK instruction at cryptographic system B verifies DSIGa. In an alternate embodiment, DSIGa can be calculated by the CF 30 using a separate instruction for generating digital signatures. In that case, after the GKSP instruction has been executed, the CFAP invokes the generate digital signature instruction causing DSIGa to be generated.

Detailed Description Text (44):
DSIG verification means 611 uses public key PU belonging to the sending cryptographic device to verify the digital signature DSIG generated on ePU(keyblk) at the sending cryptographic device. To accomplish this, a hash value is first calculated on ePU(keyblk) using hash algorithm 512. Hash algorithm 512 may in fact be a set of hash algorithms. In that case, the hash algorithm is selected on the basis of a hash algorithm identifier or other appropriate encoded value passed by the control information retrieval means 608 to the DSIG verification means (not shown in FIG. 10). The clear public key PU obtained from PU recovery means 606 is then used to encrypt the value of DSIG specified as an input at 602. This recovers the original signature block in clear form, which is then parsed to recover the original hash value. The recovered hash value and the calculated hash value are then compared for equality. If this comparison is favorable, then DSIG is considered valid; otherwise, DSIG is not considered valid and IDK instruction 600 is aborted. The signature block recovery and processing of course will depend on the method of digital signature implemented. In the description of the GKSP instruction it was indicated that the signature block may consist of the hash value and padding data or it may be constructed on the basis of a national or international standard, such as International Standards Organization draft international standard (ISO DIS) 9796. Those skilled in the art will appreciate that many possible implementations of the digital signature are possible and that the precise method of digital signatures is unimportant to the invention. What is important is that a method of digital signature is used in the preferred embodiment to ensure that the receiving cryptographic device can authenticate that the to-be-imported DEA key did in fact originate from a valid network cryptographic device. As the reader will also see, the digital signature is made an integral part of the GKSP and IDK instructions themselves, which ensures that the process of signature production and signature verification occurs as part of the key export and key import processes and therefore the highest possible integrity over these processes is achieved. Although it is possible to perform signature production and signature verification as separate instructions, which achieves complete compatibility with the present descriptions of the GKSP and IDK instructions, one also sees that less integrity is achieved. This is so because the signature generate instruction has no way to ensure that a key of the form ePU(keyblk) was in fact produced by the GKSP instruction.

Detailed Description Text (48):
The reader will appreciate that the IDK instruction has been designed to perform

consistency checking within the cryptographic facility in lieu of returning the recovered clear values of C6 and EID to the CFAP and performing this consistency checking outside of the cryptographic facility. In the preferred embodiment, this consistency checking is performed in the cryptographic facility hardware and the recovered clear values of C6 and EID are not exposed outside the CF. The reason for doing this is to ensure that the DEA key distribution channel does not also provide a covert privacy channel whereby secret data may be incorporated in the control information portion of the key block and transmitted from the sending cryptographic device to the receiving cryptographic device. In a good cryptographic design, the cryptographic instructions will perform only those cryptographic functions for which they were designed, and no more. Doing so, limits the ways in which an attacker can manipulate the cryptographic instructions for the purpose of subverting their intended security. For example, a system administrator in charge of security policy for the sending and receiving locations, may have a security policy which prohibits the transmission of private messages over the communications link, for example when the link is dedicated merely to the transmission of new keys. In an alternate security policy where the system administrator is to selectively allow privacy channels, there should be no "back door" method for subverting the system administrator's authority in enabling or prohibiting such privacy channels. The use of the control information transmitted over the separate channel to the receiver, is to enable the recipient to inspect the type of uses imposed on the receive key and allow the recipient the option of rejecting the keyblock. However, an alternate embodiment is possible wherein the recovered clear values of C6 and EID are returned to the CFAP and consistency checking is then performed by the CFAP.

CLAIMS:

1. In a data processing system having a plurality of communicating nodes, at least a pair of nodes in the system exchanging cryptographic communications, an apparatus for enabling a first node of the pair to control a crypto variable after its transmission from the first node to a second node of the pair, comprising:

a first storage means at a transmitting node in the system for storing a crypto variable which is to be transmitted to a receiving node in the system;

a second storage means at said transmitting node for storing control information to control said crypto variable after it is transmitted from said transmitting node said control information including a control vector to limit the uses of said crypto variable;

a third storage means at said transmitting node for storing a first key expression;

concatenating means at said transmitting node, coupled to said first and second storage means, for concatenating said crypto variable with said control information, forming a key block;

encryption means at said transmitting node, coupled to said third storage means and said concatenating means, for encrypting said key block with said first key expression, forming an encrypted key block;

transmitting means at said transmitting node coupled to said encryption means and coupled over a communications link to a receiving means at said receiving node, for transmitting said encrypted key block to said receiving node;

said transmitting means coupled to said second storage means, for transmitting a second copy of said control information to said receiving node;

fourth storage means at said receiving node, for storing a second key expression

corresponding to said first key expression;

decryption means at said receiving node coupled to said receiving means and to said fourth storage means, for decrypting said encrypted key block using said second key expression, to obtain a recovered key block;

extraction means at said receiving node coupled to said decryption means, to extract said control information and said crypto variable from said recovered key block;

comparison means at said receiving node coupled to said extraction means and coupled to said receiving means for comparing said control information extracted from said recovered key block to said second copy of said control information, said comparison means having an enabling output for signaling when said comparison is satisfied;

control means coupled to said extraction means and having an enabling input coupled to said output of said comparison means, for controlling said crypto variable with said control information.

3. In a processing system having a plurality of communicating nodes, at least a pair of nodes in the system exchanging cryptographic communications, a method for enabling a first node of the pair to control a crypto variable after its transmission from the first node to a second node of the pair, comprising:

storing a crypto variable which is to be transmitted to a receiving node in the system, at a transmitting node;

storing control information to control said crypto variable after it is transmitted from said transmitting node, at said transmitting node said control information including a control vector to limit the uses of said crypto variable;

storing a first key expression at said transmitting node;

concatenating said crypto variable with said control information, forming a key block, at said transmitting node;

encrypting said key block with said first key expression, forming an encrypted key block at said transmitting node;

transmitting said encrypted key block to said receiving node;

transmitting a second copy of said control information to said receiving node;

storing a second key expression corresponding to said first key expression, at said receiving node;

decrypting said encrypted key block using said second key expression, to obtain a recovered key block, at said receiving node;

extracting said control information and said crypto variable from said recovered key block, at said receiving node;

comparing said control information extracted from said recovered key block with said second copy of said control information and generating an enabling signal when the compare is satisfied;

controlling said crypto variable with said control information when said enabling signal has been generated.

13. The method of claim 12, which further comprises:

said received control vector is a first hashed product of said reference control vector, received from said transmitting node;

forming a second hash product of said reference control vector, at said receiving node;

comparing said first hashed product with said second hashed product and outputting a second acceptance signal when the comparison is satisfied, at said receiving node.

14. The method of claim 3, which further comprises: said control information includes a hashed control vector which represents limitations on uses of said crypto variable.

15. The method of claim 14, wherein said control means further comprises:

receiving from said transmitting node and storing a reference control vector characterizing required uses of said crypto variable at said receiving station, at said receiving node;

storing said hashed control vector extracted from said recovered key block, at said receiving node;

forming a hash product of said reference control vector, at said receiving node;

comparing said hash product with said hashed control vector, and outputting an acceptance signal if the comparison succeeds, at said receiving node;

storing said crypto variable extracted by said extraction means if said acceptance signal is hashed from said compare means, at said receiving node.

16. The method of claim 15, which further comprises:

storing a master key at said receiving node;

forming an exclusive OR product of said master key and said hashed control vector, forming a product key expression, at said receiving node;

inputting said product key expression, and encrypting said crypto variable under said master key, forming an encrypted crypto variable, at said receiving node;

storing said encrypted crypto variable, at said receiving node.

17. The method of claim 16, which further comprises:

receiving a request from a user for using said crypto variable, at said receiving node;

checking said reference control vector to determine if checking said reference said requested uses are permitted, at said receiving node;

outputting an enabling signal if said requested uses are permitted, at said receiving node;

receiving said enabling signal and in response thereto, forming an exclusive OR product of said master key and said hashed control vector, forming a product key expression, at said receiving node;

inputting said product key expression, and decrypting said encrypted crypto variable under said master key, recovering said crypto, variable, at said receiving node.

20. In a data processing system having a plurality of communicating nodes, at least a pair of nodes in the system exchanging cryptograpic communications, a program for execution on the data processing system for enabling a first node of the pair to control a crypto variable after its transmission from the first node to a second node of the pair, comprising:

said program controlling the data processing system for storing a crypto variable which is to be transmitted to a receiving node in the system, at a transmitting node;

said program controlling the data processing system for storing control information to control said crypto variable after it is transmitted from said transmitting node, at said transmitting node said control information including a control vector to limit the uses of said crypto variable;

said program controlling the data processing system for storing a first key expression at said transmitting node;

said program controlling the data processing system for concatenating said crypto variable with said control information, forming a key block, at said transmitting node;

said program controlling the data processing system for encrypting said key block with said first key expression, forming an encrypted key block, at said transmitting node;

said program controlling the data processing system for transmitting said encrypted key block to said receiving node;

said program controlling the data processing system for transmitting a second copy of said control information to said receiving node;

said program controlling the data processing system for storing a second key expression corresponding to said first key expression, at said receiving node;

said program controlling the data processing system for decrypting said encrypted key block using said second key expression, to obtain a recovered key block, at said receiving node;

said program controlling the data processing system for extracting said control information and said crypto variable from said recovered key block, at said receiving node;

said program controlling the data processing system for comparing said control information extracted from said recovered key bock with said second copy of said control information and generating an enabling signal when the compare is satisfied;

said program controlling the data processing system for controlling said crypto variable with said control information when said enabling signal has been generated.

24. Apparatus for generating and distributing a Data Encryption Algorithm (DEA) key in a communications network, comprising:

a) sending means for generating and producing at least two copies of a key-

encrypting key (K-ek), and control information including a control vector for permitted uses of the k-ek;

b) means included in the sending means for encrypting one copy of the k-ek under the public key of a receiving means and transmitting the public key encrypted k-ek to the receiving means in association with said control information;

c) means further included in the sending means for encrypting another copy of the k-ek under a master key of the sending means;

d) means further included in the sending means for storing the master key encrypted k-3k as a common distributing key for other encrypted keys used in the network, in association with said control information;

e) control means included in the sending means, to limit uses of the k-ek to said permitted uses in response to said control information.

Previous Doc Next Doc Go to Doc#

L14: Entry 1 of 3 File: USPT Apr 2, 2002

DOCUMENT-IDENTIFIER: US 6366950 B1
TITLE: System and method for verifying users' identity in a network using e-mail communication

Application Filing Date (1):
19990402

Detailed Description Text (7):
Although the computer 4 is connected to the computer 2, it is contemplated that the database 4 is configured to register any computer that communicates with the computer 4 and includes an identification module 8. As soon as the computer 2, or any additional computer, is registered, the computer 4 can identify and authorize the computer 2 during subsequent communications.

Detailed Description Text (14):
In accordance with one embodiment of the present invention, the ID number serves to address, identify and authorize computers. As mentioned above, the ID number is unique to a computer and cannot be altered. This provides a higher degree of reliability and security, because the IP address and the email address can be altered. For instance, some users alter the email address or the address field to camouflage the return address and, thus, their real identity.

Detailed Description Text (16):
The computer 4 receives the electronic representation of the email and converts it back to a user-readable message. During the process of converting, the computer 4 extracts the received ID number and compares (looks-up) it with the ID number(s) stored in the data base 7. When the received ID number matches one of the stored ID numbers, the computer 4 accepts the email as one received from an authorized computer.

Detailed Description Text (18):
The user of the computer 4 can define how emails from computers whose ID numbers are not stored in the database need to be treated. Depending on user-specified settings of the computer 4, emails from unauthorized/unidentified computers can be, for example, blocked or rejected. For instance, the user can create a contact list in which all authorized users are listed. If the received ID number does not match to the ID number stored for an authorized user from the contact list, the email will be rejected.

Detailed Description Text (21):
Moreover, the computer 4 cannot only block or reject emails from unauthorized users, but also identify if the return email address that appears in the field "From:" is indeed the real email address. For example, the sender of the email could pretend to be an authorized user by changing the email address to one the sender believes the computer 4 accepts. However, because the ID number is included to the received email, the false identity of the sender of the email can be recognized.

Detailed Description Text (47):
In one embodiment, the computer 20 applies a hash function to the ID number to

convert it to a first hashed ID number. In one embodiment, first hashed ID number uses the ID number and a server-specific server identifier. The server 26 then stores the first hashed ID number.

Detailed Description Text (49):
The computer 20 applies the hash function to the first hashed number, the session number, the specific key, and a second random number to convert these numbers to a second hashed ID number. That is, the second hashed ID number is a function of the hashed number, the session number, the specific key and the second random number. The computer 20 generates the second random number so that the authentication code would be different even if the session number were a fixed value.

Detailed Description Text (50):
The server 26 receives the second hashed ID number and extracts the first hashed ID number. The server 26 retrieves lie stored first hashed ID number and compares it with the extracted first hashed ID number. If the hashed ID numbers match, the computer 20 is authenticated.

Detailed Description Text (51):
The user of the computer 22 can register with the server 26 in the same way as the user of the computer 20. The identification database 32 includes then the unique ID numbers of the computers 20, 22. If both computers 20, 22 apply hash flnctions to their ID numbers, the server 26 stores two first hashed ID numbers during the registration procedure. As the their ID numbers are different, the computers 20, 22 generate different first hashed ID numbers.

Detailed Description Text (52):
In one example, the users of the computers 20, 22 have both registered with the server 26 through the procedure illustrated in FIG. 4. In addition, the computers 20, 22 defined contact lists so that the computers 20, 22 accept only emails from authorized computers.

CLAIMS:

1. A method of maintaining a user identification database that indicates when users are in communication with a network, the method comprising the acts of:

associating in a computer accessible storage medium electronic mail addresses, processor-embedded identifiers and status information;

receiving a first electronic message from a first computer, the first electronic message containing an electronic mail address and a copy of the processor embedded identifier existing in the first computer;

using the first electronic mail address to access the corresponding processor-embedded identifier stored in the storage medium;

comparing the processor-embedded identifier from the first computer with the processor-embedded identifiers of the storage medium;

modifying the status information in the storage medium to indicate that the first electronic mail address is authentic when the processor-embedded identifier from the first computer matches a processor-embedded identifier of storage medium;

receiving a second electronic message from a second computer, the second electronic message requesting authentication of the first electronic mail address;

comparing the first electronic mail address with the electronic mail addresses stored in the storage medium; and

sending a third message to the second computer that indicates whether the first electronic mail address is authentic.

10. The method of claim 8, wherein the act of altering includes hashing the processor-embedded identifier.

Previous Doc Next Doc Go to Doc#

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 7 of 7 returned.

☐ 1. Document ID: US 6782475 B1

L17: Entry 1 of 7 File: USPT Aug 24, 2004

US-PAT-NO: 6782475
DOCUMENT-IDENTIFIER: US 6782475 B1

TITLE: Method and apparatus for conveying a private message to selected members

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 2. Document ID: US 6701434 B1

L17: Entry 2 of 7 File: USPT Mar 2, 2004

US-PAT-NO: 6701434
DOCUMENT-IDENTIFIER: US 6701434 B1

TITLE: Efficient hybrid public key signature scheme

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 3. Document ID: US 6339828 B1

L17: Entry 3 of 7 File: USPT Jan 15, 2002

US-PAT-NO: 6339828
DOCUMENT-IDENTIFIER: US 6339828 B1

TITLE: System for supporting secured log-in of multiple users into a plurality of computers using combined presentation of memorized password and transportable passport record

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 4. Document ID: US 6061790 A

L17: Entry 4 of 7 File: USPT May 9, 2000

US-PAT-NO: 6061790
DOCUMENT-IDENTIFIER: US 6061790 A

h e b b g e e e f e f ef b e

TITLE: Network computer system with remote user data encipher methodology

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 5. Document ID: US 6052787 A

L17: Entry 5 of 7 File: USPT Apr 18, 2000

US-PAT-NO: 6052787
DOCUMENT-IDENTIFIER: US 6052787 A

TITLE: Process for group-based cryptographic code management between a first computer unit and group computer units

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 6. Document ID: US 5717757 A

L17: Entry 6 of 7 File: USPT Feb 10, 1998

US-PAT-NO: 5717757
DOCUMENT-IDENTIFIER: US 5717757 A

TITLE: Certificate issue lists

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

☐ 7. Document ID: US 5265164 A

L17: Entry 7 of 7 File: USPT Nov 23, 1993

US-PAT-NO: 5265164
DOCUMENT-IDENTIFIER: US 5265164 A

TITLE: Cryptographic facility environment backup/restore and replication in a public key cryptosystem

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

---

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L16 not L14	7

Display Format: TI Change Format

h e b b g e e e f e f ef b e

Previous Page Next Page Go to Doc#

h e b b g e e e f e f ef b e

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 10 of 20 returned.

☐ 1. Document ID: JP 2000306328 A

Using default format because multiple data bases are involved.

L19: Entry 1 of 20 File: JPAB Nov 2, 2000

PUB-NO: JP02000306328A
DOCUMENT-IDENTIFIER: JP 2000306328 A
TITLE: APPARATUS AND METHOD FOR PROCESSING INFORMATION AND PROGRAM STORING MEDIUM

PUBN-DATE: November 2, 2000

INVENTOR-INFORMATION:

NAME	COUNTRY
KAWAKAMI, TATSU	
ISHIGURO, RYUJI	
TANABE, MITSURU	
EOMO, YUICHI	

INT-CL (IPC): G11 B 20/10

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 2. Document ID: JP 2000305846 A

L19: Entry 2 of 20 File: JPAB Nov 2, 2000

PUB-NO: JP02000305846A
DOCUMENT-IDENTIFIER: JP 2000305846 A
TITLE: INFORMATION PROCESSOR, ITS METHOD AND PROGRAM STORING MEDIUM

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 3. Document ID: JP 2000227757 A

L19: Entry 3 of 20 File: JPAB Aug 15, 2000

PUB-NO: JP02000227757A
DOCUMENT-IDENTIFIER: JP 2000227757 A
TITLE: WEB PAGE AUTHENTICATION SYSTEM

h e b b g e e e f e f ef b e

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 4. Document ID: JP 11175512 A

L19: Entry 4 of 20 File: JPAB Jul 2, 1999

PUB-NO: JP411175512A
DOCUMENT-IDENTIFIER: JP 11175512 A
TITLE: PROGRAM RELATED WITH PRESENCE CERTIFICATION OF DOCUMENT

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 5. Document ID: JP 09297828 A

L19: Entry 5 of 20 File: JPAB Nov 18, 1997

PUB-NO: JP409297828A
DOCUMENT-IDENTIFIER: JP 09297828 A
TITLE: AUTHENTICATION TYPE SECURITY SYSTEM

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 6. Document ID: JP 09245139 A

L19: Entry 6 of 20 File: JPAB Sep 19, 1997

PUB-NO: JP409245139A
DOCUMENT-IDENTIFIER: JP 09245139 A
TITLE: AUTHENTICATING TYPE SECURITY SYSTEM

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 7. Document ID: JP 06012303 A

L19: Entry 7 of 20 File: JPAB Jan 21, 1994

PUB-NO: JP406012303A
DOCUMENT-IDENTIFIER: JP 06012303 A
TITLE: METHOD FOR JUDGING WHETHER OR NOT RECORD IS STORED IN COMPUTER SYSTEM AND DEVICE FOR THE SAME

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 8. Document ID: WO 9625801 A1

L19: Entry 8 of 20 File: EPAB Aug 22, 1996

PUB-NO: WO009625801A1
DOCUMENT-IDENTIFIER: WO 9625801 A1

h e b b g e e e f e f ef b e

TITLE: METHOD FOR PARTITIONING A BLOCK OF DATA INTO SUBBLOCKS AND FOR STORING AND COMMUNICATING SUCH SUBBLOCKS

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

☐ 9. Document ID: EP 117281 A2

L19: Entry 9 of 20 File: EPAB Sep 5, 1984

PUB-NO: EP000117281A2
DOCUMENT-IDENTIFIER: EP 117281 A2
TITLE: Updating data processing files.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

☐ 10. Document ID: NNRD425106

L19: Entry 10 of 20 File: TDBD Sep 1, 1999

TDB-ACC-NO: NNRD425106

DISCLOSURE TITLE: Scheme for Restricting Execution of Unlicensed or Virus-Infected Software on a Hardware Platform (Virus and Pirated Software Resistant System)

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1999. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L18 and (copy or copying or duplicat$ or copied)	20

Display Format: - Change Format

Previous Page Next Page Go to Doc#

h e b b g e e e f e f ef b e

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 11 through 20 of 20 returned.

☐ 11. Document ID: NN9509127

Using default format because multiple data bases are involved.

L19: Entry 11 of 20 File: TDBD Sep 1, 1995

TDB-ACC-NO: NN9509127

DISCLOSURE TITLE: Document Repository

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, September 1995, US

VOLUME NUMBER: 38
ISSUE NUMBER: 9
PAGE NUMBER: 127 - 128

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1995. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

☐ 12. Document ID: NN9409453

L19: Entry 12 of 20 File: TDBD Sep 1, 1994

TDB-ACC-NO: NN9409453

DISCLOSURE TITLE: High Integrity Distributed Configuration Management

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1994. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

h e b b cg b cc e

☐ 13. Document ID: NN9403413

L19: Entry 13 of 20 File: TDBD Mar 1, 1994

TDB-ACC-NO: NN9403413

DISCLOSURE TITLE: Multimedia Mixed Object Envelopes Supporting a Graduated Fee Scheme via Encryption

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1994. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Drawa De

---

☐ 14. Document ID: NB9209201

L19: Entry 14 of 20 File: TDBD Sep 1, 1992

TDB-ACC-NO: NB9209201

DISCLOSURE TITLE: Dynamic Method Name Registration for Improved Search Speed.

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1992. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Drawa De

---

☐ 15. Document ID: NN8710361

L19: Entry 15 of 20 File: TDBD Oct 1, 1987

TDB-ACC-NO: NN8710361

DISCLOSURE TITLE: PARALLEL EQUI-JOIN ALGORITHM for LARGE RELATIONAL DATA BASE Operations

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1987. All rights reserved.

h e b b cg b cc e

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

16. Document ID: NN84046046

L19: Entry 16 of 20 File: TDBD Apr 1, 1984

TDB-ACC-NO: NN84046046

DISCLOSURE TITLE: Extendible Hashing for Line-Oriented Paging Stores

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1984. All rights reserved.

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

17. Document ID: TW 518455 A, WO 200004713 A1, EP 1018266 A1, CN 1274504 A, KR 2001024179 A, JP 2002521876 W, US 6530021 B1

L19: Entry 17 of 20 File: DWPI Jan 21, 2003

DERWENT-ACC-NO: 2000-223727
DERWENT-WEEK: 200356
COPYRIGHT 2004 DERWENT INFORMATION LTD

TITLE: Method for preventing unauthorized playback of digital data streams involves applying cryptographic hashing function to saved ticket which is then compared to extracted watermark

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

18. Document ID: WO 9625801 A1, US 5990810 A, AU 9646593 A

L19: Entry 18 of 20 File: DWPI Aug 22, 1996

DERWENT-ACC-NO: 1996-393645
DERWENT-WEEK: 200002
COPYRIGHT 2004 DERWENT INFORMATION LTD

TITLE: Data block partitioning method for e.g document communication - in which blocks are divided into fixed-length sub-blocks which are compared with each other to determine identical blocks

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

19. Document ID: US 5390359 A

h e b b cg b cc e

L19: Entry 19 of 20 File: DWPI Feb 14, 1995

DERWENT-ACC-NO: 1995-090542
DERWENT-WEEK: 199512
COPYRIGHT 2004 DERWENT INFORMATION LTD

TITLE: Record storage and retrieval method for computer system - applying hash function to subsets of key representing record to be stored to generate multiple hash addresses whenever record is stored, and storing copy of key or pointer at hash addresses

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

20. Document ID: WO 9114992 A, AU 9174708 A, JP 05505050 W, US 5129082 A

L19: Entry 20 of 20 File: DWPI Oct 3, 1991

DERWENT-ACC-NO: 1991-310729
DERWENT-WEEK: 199142
COPYRIGHT 2004 DERWENT INFORMATION LTD

TITLE: Search and retrieval appts. for database component files - uses database component file with unique name dependent upon file contents with name changing when source file contents change

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L18 and (copy or copying or duplicat$ or copied)	20

Display Format: - Change Format

Previous Page Next Page Go to Doc#

h e b b cg b cc e

First Hit Previous Doc Next Doc Go to Doc#
End of Result Set

Generate Collection Print

L20: Entry 1 of 1 File: TDBD Sep 1, 1999

TDB-ACC-NO: NNRD425106

DISCLOSURE TITLE: Scheme for Restricting Execution of Unlicensed or Virus-Infected Software on a Hardware Platform (Virus and Pirated Software Resistant System)

PUBLICATION-DATA:
Research Disclosure, September 1999, UK

VOLUME NUMBER: 42
ISSUE NUMBER: 425
PUBLICATION-DATE: September 1, 1999 (19990901)

CROSS REFERENCE: 0374-4353-42-425-0

DISCLOSURE TEXT:

THIS COPY WAS MADE FROM AN INTERNAL IBM DOCUMENT AND NOT FROM THE PUBLISHED BOOK
One problem that administrators and IT organizations have today is not having control over what software is being run on the systems they administrate. This can lead to legal problems of pirated software running on corporate systems, or viruses being introduced into PCs on a manufacturing line by an employee who brings in an infected game to run on a computer on the manufacturing line or it can lead to unauthorized use of systems. In those cases it is important to be able to restrict the programs that can run on a particular platform to those that are approved. This invention describes how that can be accomplished.

SECURITY: Use, copying and distribution of this data is subject to the restictions in the Agreement For IBM TDB Database and Related Computer Databases. Unpublished - all rights reserved under the Copyright Laws of the United States. Contains confidential commercial information of IBM exempt from FOIA disclosure per 5 U.S.C. 552(b)(4) and protected under the Trade Secrets Act, 18 U.S.C. 1905.

COPYRIGHT STATEMENT: The text of this article is Copyrighted (c) IBM Corporation 1999. All rights reserved.

Previous Doc Next Doc Go to Doc#

L14: Entry 2 of 3 File: USPT Sep 18, 2001

DOCUMENT-IDENTIFIER: US 6292569 B1
**** See image for Certificate of Correction ****
TITLE: Systems and methods using cryptography to protect secure computing environments

Abstract Text (1):
Secure computation environments are protected from bogus or rogue load modules, executables and other data elements through use of digital signatures, seals and certificates issued by a verifying authority. A verifying authority--which may be a trusted independent third party--tests the load modules or other executables to verify that their corresponding specifications are accurate and complete, and then digitally signs the load module or other executable based on tamper resistance work factor classification. Secure computation environments with different tamper resistance work factors use different verification digital signature authentication techniques (e.g., different signature algorithms and/or signature verification keys)--allowing one tamper resistance work factor environment to protect itself against load modules from another, different tamper resistance work factor environment. Several dissimilar digital signature algorithms may be used to reduce vulnerability from algorithm.

Application Filing Date (1):
20001004

Brief Summary Text (2):
This invention relates to computer security, and more particularly to secure and/or protected computer execution environments. Still more specifically, the present invention relates to computer security techniques based at least in part on cryptography, that protect a computer processing environment against potentially harmful computer executables, programs and/or data; and to techniques for certifying load modules such as executable computer programs or fragments thereof as being authorized for use by a protected or secure processing environment.

Brief Summary Text (20):
the owner may wish to defeat other electronic controls preventing him or her from performing certain tasks (for example, copying content without authorization); or

Brief Summary Text (34):
In accordance with one aspect provided by the present invention, one or more trusted verifying authorities validate load modules or other executables by analyzing and/or testing them. A verifying authority digitally "signs" and "certifies" those load modules or other executables it has verified (using a public key based digital signature and/or certificate based thereon, for example).

Brief Summary Text (35):
Protected execution spaces such as protected processing environments can be programmed or otherwise conditioned to accept only those load modules or other executables bearing a digital signature/certificate of an accredited (or particular) verifying authority. Tamper resistant barriers may be used to protect this programming or other conditioning. The assurance levels described below are a measure or assessment of the effectiveness with which this programming or other

conditioning is protected.

Brief Summary Text (36):
A web of trust may stand behind a verifying authority. For example, a verifying authority may be an independent organization that can be trusted by all electronic value chain participants not to collaborate with any particular participant to the disadvantage of other participants. A given load module or other executable may be independently certified by any number of authorized verifying authority participants. If a load module or other executable is signed, for example, by five different verifying authority participants, a user will have (potentially) a higher likelihood of finding one that they trust. General commercial users may insist on several different certifiers, and government users, large corporations, and international trading partners may each have their own unique "web of trust" requirements. This "web of trust" prevents value chain participants from conspiring to defraud other value chain participants.

Brief Summary Text (38):
A verifying authority analyzes, validates, verifies, inspects, and/or tests the load module or other executable, and compares its results with the specifications associated with the load module or other executable. A verifying authority may digitally sign or certify only those load modules or other executables having proper specifications--and may include the specifications as part of the material being signed or certified.

Brief Summary Text (39):
A verifying authority may instead, or in addition, selectively be given the responsibility for analyzing the load module and generating a specification for it. Such a specification could be reviewed by the load module's originator and/or any potential users of the load module.

Brief Summary Text (40):
A verifying authority may selectively be given the authority to generate an additional specification for the load module, for example by translating a formal mathematical specification to other kinds of specifications. This authority could be granted, for example, by a load module originator wishing to have a more accessible, but verified (certified), description of the load module for purposes of informing other potential users of the load module.

Brief Summary Text (41):
Additionally, a verifying authority may selectively be empowered to modify the specifications to make it accurate--but may refuse to sign or certify load modules or other executables that are harmful or dangerous irrespective of the accuracy of their associated specifications. The specifications may in some instances be viewable by ultimate users or other value chain participants--providing a high degree of assurance that load modules or other executables are not subverting the system and/or the legitimate interest of any participant in an electronic value chain the system supports.

Brief Summary Text (45):
A verifying authority may digitally sign load modules or other executables with a digital signature that indicates or implies assurance level. A verifying authority can use digital signature techniques to distinguish between assurance levels. As one example, each different digital signature may be encrypted using a different verification key and/or fundamentally different encryption, one-way hash and/or other techniques. A protected processing environment or other secure execution space protects itself by executing only those load modules or other executables that have been digitally signed for its corresponding assurance level.

Brief Summary Text (46):
The present invention may use a verifying authority and the digital signatures it

provides to compartmentalize the different electronic appliances depending on their level of security (e.g., work factor or relative tamper resistance). In particular, a verifying authority and the digital signatures it provides isolate appliances with significantly different work factors--preventing the security of high work factor appliances from collapsing into the security of low work factor appliances due to free exchange of load modules or other executables.

Brief Summary Text (48):
In accordance with another aspect provided by this invention, a verifying authority can digitally sign a load module or other executable with several different digital signatures and/or signature schemes. A protected processing environment or other secure execution space may require a load module or other executable to present multiple digital signatures before accepting it. An attacker would have to "break" each (all) of the several digital signatures and/or signature schemes to create an unauthorized load module or other executable that would be accepted by the protected processing environment or other secure execution space. Different protected processing environments (secure execution spaces) might examine different subsets of the multiple digital signatures--so that compromising one protected processing environment (secure execution space) will not compromise all of them. As an optimization, a protected processing environment or other secure execution space might verify only one of the several digital signatures (for example, chosen at random each time an executable is used)--thereby speeding up the digital signature verification while still maintaining a high degree of security.

Drawing Description Text (4):
FIG. 2 shows an example verification authority that protects the electronic community from unauthorized load modules;

Drawing Description Text (5):
FIG. 3 shows how a protected processing environment can distinguish between load modules that have been approved by a verifying authority and those that have not been approved;

Drawing Description Text (6):
FIG. 4 shows an example process a verifying authority may perform to authenticate load modules;

Drawing Description Text (7):
FIG. 5 shows how a verifying authority can create a certifying digital signature;

Drawing Description Text (8):
FIG. 6 shows how a protected processing environment can securely authenticate a verifying authority's digital signature to guarantee the integrity of the corresponding load module;

Drawing Description Text (13):
FIGS. 10A-10C show how different assurance level electronic appliances can be provided with different cryptographic keys for authenticating verifying authority digital signatures;

Drawing Description Text (14):
FIGS. 11A-11C show how a verifying authority can use different digital signatures to designate the same or different load modules as being appropriate for execution by different assurance level electronic appliances;

Detailed Description Text (2):
FIG. 1 shows how defective, bogus and/or unauthorized computer information can wreak havoc within an electronic system 50. In this example, provider 52 is authorized to produce and distribute "load modules" 54 for use by different users or consumers 56. FIG. 1 shows "load module" 54 as a complicated looking machine

part for purposes of illustration only; the load module preferably comprises one or more computer instructions and/or data elements used to assist, allow, prohibit, direct, control or facilitate at least one task performed at least in part by an electronic appliance such as a computer. For example, load module 54 may comprise all or part of an executable computer program and/or associated data ("executable"), and may constitute a sequence of instructions or steps that bring about a certain result within a computer or other computation element.

Detailed Description Text (7):
FIG. 1 also shows an unauthorized and/or disreputable load module provider 64. Unauthorized provider 64 knows how to make load modules that look a lot like the load modules produced by authorized load module provider 52--but are defective or even destructive. Unless precautions are taken, the unauthorized load module 54d made by unauthorized producer 64 will be able to run on protected processing environments 108 within appliances 58, 60 and 62, and may cause serious harm to users 56 and/or to the integrity of system 50. For example:

Detailed Description Text (8):
Unauthorized provider 64 could produce a load module 54d that is quite similar to authorized load module 54a intended to be used by set top box or home media player 58. The unauthorized load module 54d might allow protected processing environment 108A within set top box/home media player 58 to present the very same program material --but divert some or all of the user's payment to unauthorized producer 64--thereby defrauding the rights holders in the program material the users watch.

Detailed Description Text (11):
FIG. 2 shows how a verifying authority 100 can prevent the problems shown in FIG. 1. In this example, authorized provider 52 submits load modules 54 to verifying authority 100. Verifying authority 100 carefully analyzes the load modules 54 (see 102), testing them to make sure they do what they are supposed to do and do not compromise or harm system 50. If a load module 54 passes the tests verifying authority 100 subjects it to, a verifying authority may affix a digital "seal of approval" (see 104) to the load module.

Detailed Description Text (12):
Protected processing environments 108 can use this digital "seal of approval" 106 (which may comprise one or more "digital signatures") to distinguish between authorized and unauthorized load modules 54. FIG. 3 illustrates how an electronic protected processing environment 108 can use and rely on a verifying authority's digital seal of approval 106. In this example, the protected processing environment 108 can distinguish between authorized and unauthorized load modules 54 by examining the load module to see whether it bears the seal of verifying authority 100. Protected processing environment 108 will execute the load module 54a with its processor 110 only if the load module bears a verifying authority's seal 106. Protected processing environment 108 discards and does not use any load module 54 that does not bear this seal 106. In this way, protected processing environment 108 securely protects itself against unauthorized load modules 54 such as, for example, the defective load module 54d made by disreputable load module provider 64.

Detailed Description Text (13):
FIG. 4 shows the analysis and digital signing steps 102, 104 performed by verifying authority 100 in this example. Provider 54 may provide, with each load module 54, associated specifications 110 identifying the load module and describing the functions the load module performs. In this example, these specifications 110 are illustrated as a manufacturing tag, but preferably comprises a data file associated with and/or attached to the load module 54.

Detailed Description Text (14):
Verifying authority 100 uses an analyzing tool(s) 112 to analyze and test load module 54 and determine whether it performs as specified by its associated

specifications 110--that is, whether the specifications are both accurate and complete. FIG. 4 illustrates an analysis tool 112 as a magnifying glass; verifying authority 100 may not rely on visual inspection only, but instead preferably uses one or more computer-based software testing techniques and/or tools to verify that the load module performs as expected, matches specifications 110, is not a "virus," and includes no significant detectable "bugs" or other harmful functionality. See for example Pressman, Software Engineering: A Practitioner's Approach (3d Ed., McGraw-Hill 1992) at chapters 18 and 19 ("Software Testing Techniques") (pages 595-661) and the various books and papers referenced there. Although it has been said that "testing can show only the presence of bugs, not their absence," such testing (in addition to ensuring that the load module 54 satisfies its specifications 110) can provide added degrees of assurance that the load module isn't harmful and will work as it is supposed to.

Detailed Description Text (15):
Verifying authority 100 is preferably a trusted, independent third party such as an impartial, well respected independent testing laboratory. Therefore, all participants in an electronic transaction involving load module 54 can trust a verifying authority 100 as performing its testing and analysis functions competently and completely objectively and impartially. As described above, there may be several different verifying authorities 100 that together provide a "web of trust". Several different verifying authorities may each verify and digitally sign the same load module--increasing the likelihood that a particular value chain participant will trust one of them and decreasing the likelihood of collusion or fraud. Electronic value chain participants may rely upon different verifying authorities 100 to certify different types of load modules. For example, one verifying authority 100 trusted by and known to financial participants might verify load modules relating to financial aspects of a transaction (e.g., billing), whereas another verifying authority 100' trusted by and known to participants involved in using the "information exhaust" provided by an electronic transaction might be used to verify load modules relating to usage metering aspects of the same transaction.

Detailed Description Text (16):
Once verifying authority 100 is satisfied with load module 54, it affixes its digital "seal of approval" 106 to the load module. FIG. 4 illustrates the digital sealing process as being performed by a stamp 114--but in the preferred embodiment the digital sealing process is actually performed by creating a "digital signature" using a well known process. See Schneier, Applied Cryptography (2d Ed. John Wiley & Sons 1996) at Chapter 20 (pages 483-502). This digital signature, certificate or seal creation process is illustrated in FIG. 5.

Detailed Description Text (19):
In this case the first key is owned by verifying authority 100 and is kept highly secure (for example, using standard physical and procedural measures typically employed to keep an important private key secret while preventing it from being lost). Once message digest 116 is locked into strong box 118 using the first key 122 the strong box can be opened only by using the corresponding second key 124. Note that other items (e.g., further identification information, a time/date stamp, etc.) can also be placed within strong box 106.

Detailed Description Text (23):
The load module 54 and its associated digital signature 106 is then delivered to the protected processing environment 108. (These items can be provided together at the same time, independently, or at different times.) Protected processing environment 115 applies the same one way hash transformation on load module 54 that a verifying authority 100 applied. Since protected processing environment 108 starts with the same load module 54 and uses the same one-way hash function 115, it should generate the same message digest 116'.

Detailed Description Text (24):
Protected processing environment 108 then decrypts digital signature 106 using the second key 124--i.e., it opens strongbox 118 to retrieve the message digest 116 a verifying authority 100 placed in there. Protected processing environment 108 compares the version of message digest 116 it obtains from the digital signature 106 with the version of message digest 116' it calculates itself from load module 54 using the one way hash transformation 115. The message digests 116, 116' should be identical. If they do not match, digital signature 106 is not authentic or load module 54 has been changed--and protected processing environment 108 rejects load module 54.

Detailed Description Text (32):
Multiple signatures as shown in FIG. 8 impose a cost of additional storage for the signatures 106 in each protected load module 54, additional code in the protected processing environment 108 to implement additional algorithms, and additional time to verify the digital signatures (as well as to generate them at verification time). As an optimization to the use of multiple keys or algorithms, an appliance 61 might verify only a subset of several signatures associated with a load module 54 (chosen at random) each time the load module is used. This would speed up signature verification while maintaining a high probability of detection. For example, suppose there are one hundred "private" verification keys, and each load module 54 carries one hundred digital signatures. Suppose each protected processing environment 108, on the other hand, knows only a few (e.g., ten) of these corresponding "public" verification keys randomly selected from the set. A successful attack on that particular protected processing environment 108 would permit it to be compromised and would also compromise any other protected processing environment possessing and using precisely that same set of ten keys. However, it would not compromise most other protected processing environments--since they would employ a different subset of the keys used by verifying authority 100.

Detailed Description Text (38):
These three signatures 55(1), 55(2), 55(3) could all be affixed by the same verifying authority 100, or they could be affixed by three different verifying authorities (providing a "web of trust"). (In another model, a load module is verified in its entirety by multiple parties--if a user trusts any of them, she can trust the load module.) A protected processing environment 108 would need to have all three corresponding "public" keys 124(1), 124(2), 124(3) to authenticate the entire load module 54--or the different load module segments could be used by different protected processing environments possessing the corresponding different keys 124(1), 124(2), 124(3). Different signatures 55(1), 55(2), 55(3) could be calculated using different signature and/or one-way hash algorithms to increase the difficulty of defeating them by cryptanalytic attack.

Detailed Description Text (40):
Verifying authority 100 can use different digital signing techniques to provide different "assurance levels" for different kinds of electronic appliances 61 having different "work factors" or levels of tamper resistance. FIGS. 10A-10C show an example assurance level hierarchy providing three different assurance levels for different electronic appliance types:

Detailed Description Text (44):
In this example, verifying authority 100 digitally signs load modules 54 using different digital signature techniques (for example different "private" keys 122) based on assurance level. The digital signatures 106 applied by verifying authority 100 thus securely encode the same (or different) load module 54 for use by appropriate corresponding assurance level electronic appliances 61.

Detailed Description Text (46):
Within a protected processing environment 108, as shown in FIGS. 10A-10C, different

assurance levels may be assigned to each separate instance of a channel (see Ginter et al., FIG. 15) contained therein. In this way, each secure processing environment and host event processing environment (see Ginter et al., FIG. 10 and associated description) contained within an instance of a PPE 108 may contain multiple instances of a channel, each with independent and different assurance levels. The nature of this feature of the invention permits the separation of different channels within a PPE 108 from each other, each channel possibly having identical, shared, or independent sets of load modules for each specific channel limited solely to the resources and services authorized for use by that specific channel. In this way, the security of the entire PPE is enhanced and the effect of security breaches within each channel is compartmentalized solely to that channel.

Detailed Description Text (51):
In this example, electronic appliances 61 of different assurance levels can communicate with one another and pass load modules 54 between one another--an important feature providing a scaleable virtual distribution environment involving all sorts of different appliances (e.g., personal computers, laptop computers, handheld computers, television sets, media players, set top boxes, internet browser appliances, smart cards, mainframe computers, etc.) The present invention uses verifying authority 100 and the digital signatures it provides to compartmentalize the different electronic appliances depending on their level of security (e.g., work factor or relative tamper resistance). In particular, verifing authority 100 and the digital signatures it provides isolate appliances with significantly different work factors--preventing the security of high work factor appliances from collapsing into the security of low work factor appliances due to free exchange of load modules 54.

Detailed Description Text (52):
In one example, verifying authority 100 may digitally sign identical copies of load module 54 for use by different classes or "assurance levels" of electronic appliances 61. If the sharing of a load module 54 between different electronic appliances is regarded as an open communications channel between the protected processing environments 108 of the two appliances, it becomes apparent that there is a high degree of risk in permitting such sharing to occur. In particular, the extra security assurances and precautions of the more trusted environment are collapsed into the those of the less trusted environment because an attacker who compromises a load module within a less trusted environment is then be able to launch the same load module to attack the more trusted environment. Hence, although compartmentalization based on encryption and key management can be used to restrict certain kinds of load modules 54 to execute only on certain types of electronic appliances 61, a significant application in this context is to compartmentalize the different types of electronic appliances and thereby allow an electronic appliance to protect itself against load modules 54 of different assurance levels.

Detailed Description Text (66):
In accordance with this feature of the invention, verifying authority 100 supports all of these various categories of digital signatures, and system 50 uses key management to distribute the appropriate verification keys to different assurance level devices. For example, verifying authority 100 may digitally sign a particular load module 54 such that only hardware-only based server(s) 402(3) at assurance level XI may authenticate it. This compartmentalization prevents any load module executable on hardware-only servers 402(3) from executing on any other assurance level appliance (for example, software-only protected processing environment based support service 404(1)).

Detailed Description Text (71):
FIG. 14 shows an example sequence of steps that may be performed in an overall process provided by these inventions. To begin the overall process, a load module provider 52 may manufacture a load module and associated specifications (FIG. 14, block 502). Provider 52 may then submit the load module and associated

specifications to verifying authority 100 for verification (FIG. 14, block 504). Verifying authority 100 may analyze, test, and/or otherwise validate the load module against the specifications (FIG. 14, block 506), and determine whether the load module satisfies the specifications.

Detailed Description Text (72):
If the load module is found to satisfy its specifications, a verifying authority 100 determines whether it is authorized to generate one or more new specifications for the load module (FIG. 14, block 509). If it is authorized and this function has been requested ("Y" exit to decision block 509), a verifying authority generates specifications and associates them with the load module (FIG. 14, block 514).

Detailed Description Text (73):
If the load module fails the test ("N" exit to decision block 508), verifying authority 100 determines whether it is authorized and able to create new specifications corresponding to the actual load module performance, and whether it is desirable to create the conforming specifications (FIG. 14, decision block 510). If verifying authority 100 decides not to make new specifications ("N" exit to decision block 510), verifying authority returns the load module to provider 52 (block 512) and the process ends. On the other hand, if verifying authority 100 determines that it is desirable to make new specifications and it is able and authorized to do so, a verifying authority 100 may make new specifications that conform to the load module ("Y" exit to decision block 510; block 514).

Detailed Description Text (74):
A verifying authority 100 may then digitally sign the load module 54 to indicate approval (FIG. 14, block 516). This step 516 may involve applying multiple digital signatures and/or a selection of the appropriate digital signatures to use in order to restrict the load module to particular "assurance levels" of electronic appliances as discussed above. Verifying authority may then determine the distribution of the load module (FIG. 14, block 518). This "determine distribution" step may involve, for example, determining who the load module should be distributed to (e.g., provider 52, support services 404, a load module repository operated by a verifying authority, etc.) and/or what should be distributed (e.g., the load module plus corresponding digital signatures, digital signatures only, digital signatures and associated description, etc.). Verifying authority 100 may then distribute the appropriate information to a value chain using the appropriate distribution techniques (FIG. 14, block 520).

CLAIMS:

2. A method including the following:

generating an executable program;

providing the program to a certification authority;

at the certification authority:

creating a first hash based on at least a portion of the program, the hash being based on a hashing algorithm;

selecting a security level, the security level being based on an attribute of the program or of the environment in which at least a portion of the program was modified or tested;

based at least in part on the selected security level, choosing a set of keys;

using a first key from the set of keys to digitally encrypt the hash so as to create a first digital signature;

using a second key from the set of keys to digitally encrypt the hash so as to create a second digital signature, the second key being different from the first key;

associating the first digital signature and the second digital signature with the program;

providing the program to a first electronic appliance, the first electronic appliance having a security level;

at the first electronic appliance:

determining whether the security level used by the certification authority for choosing the key set is compatible with the security level present at the first electronic appliance;

if the security level used by the certification authority is compatible with the first electronic appliance security level, using a third key stored at the first electronic appliance to decrypt the first digital signature, thereby retrieiving the first hash, the third key being a copy of or otherwise associated with the first key;

using the hashing algorithm to create a second hash of at least a portion of the program;

comparing the first hash to the second hash; and

if the first hash and the second hash are identical, then proceeding with a use of the program.

3. A method as in claim 2, further including:

generating a specification which at least in part describes the functioning of the program;

providing the specification to the certification authority;

at the certification authority, prior to the step of creating the first hash, comparing the program to the specification;

if the comparison fails, generating an indication that the program does not match the specification, and ending the process without creating the first hash or performing other steps; and

if the comparison succeeds, proceeding with the step of creating the first hash.

4. A method as in claim 3, further including:

at the certification authority:

using the hashing algorithm to create a third hash, the third hash being based on at least a portion of the specification,

prior to the use of the first key set to encrypt the first hash, combining the first hash and the third hash, so that the encryption steps apply to both the first hash and the third hash;

the step of providing the program to the first electronic appliance further includes providing the specification to the first electronic appliance; and

at the first electronic appliance:

the step of retrieving the first hash includes also retrieving the third hash;

using the hashing algorithm to create a fourth hash of at least a portion of the specification, and

comparing the third hash and the second hash.

5. A method as in claim 2, further including:

providing the program to a second electronic appliance, the second electronic appliance having a security level;

at the second electronic appliance:

determining whether the security level used by the certification authority for choosing the key set is compatible with the security level present at the second electronic appliance;

if the security level used by the certification authority is compatible with the second electronic appliance security level, using a fourth key stored at the first electronic appliance to decrypt the second digital signature, thereby retrieiving the first hash, the fourth key being a copy of or otherwise associated with the second key;

using the hashing algorithm to create a second hash of at least a portion of the program,

comparing the first hash to the second hash; and

if the first hash and the second hash are identical, then proceeding with a use of the program.

9. A method including the following:

generating a computer program;

providing the computer program to a first certification authority;

at the first certification authority:

generating a first digital certificate containing information relating to the computer program,

associating the first digital certificate with the computer program, and

providing the computer program to a second certification authority; at the second certification authority:

generating a second digital certificate containing information relating to the computer program, and

associating the second digital certificate with the computer program; providing the computer program to an electronic appliance; and at the electronic appliance:

selecting whether to use the first digital certificate or the second digital certificate, the selection resulting in a decision to use the first digital certificate,

processing the first digital certificate to recover information relating to the computer program, and

using the first digital certificate information to determine whether to make a use of the computer program.

12. A method as in claim 9, in which:

generating the first digital certificate includes incorporating information about the first certification authority.

13. A method as in claim 12, in which:

the information recovered by the electronic appliance in the processing step includes the information about the first certification authority, and

the use made of the recovered information by the electronic appliance includes determining the extent to which the electronic appliance is authorized to accept digital certificates from the first certification authority.

Previous Doc Next Doc Go to Doc#